

Sinus Endoscopy - Application of Advanced GPU Volume Rendering for Virtual Endoscopy

Arno Krüger, Christoph Kubisch, Gero Strauß, and Bernhard Preim

Abstract—For difficult cases in endoscopic sinus surgery, a careful planning of the intervention is necessary. Due to the reduced field of view during the intervention, the surgeons have less information about the surrounding structures in the working area compared to open surgery. Virtual endoscopy enables the visualization of the operating field and additional information, such as risk structures (e.g., optical nerve and skull base) and target structures to be removed (e.g., mucosal swelling). The Sinus Endoscopy system provides the functional range of a virtual endoscopic system with special focus on a realistic representation. Furthermore, by using direct volume rendering, we avoid time-consuming segmentation steps for the use of individual patient datasets. However, the image quality of the endoscopic view can be adjusted in a way that a standard computer with a modern standard graphics card achieves interactive frame rates with low CPU utilization. Thereby, characteristics of the endoscopic view are systematically used for the optimization of the volume rendering speed. The system design was based on a careful analysis of the endoscopic sinus surgery and the resulting needs for computer support. As a small standalone application it can be instantly used for surgical planning and patient education. First results of a clinical evaluation with ENT surgeons were employed to fine-tune the user interface, in particular to reduce the number of controls by using appropriate default values wherever possible. The system was used for preoperative planning in 102 cases, provides useful information for intervention planning (e.g., anatomic variations of the Rec. Frontalis), and closely resembles the intraoperative situation.

Index Terms—medical visualization, sinus surgery, operation planning, virtual endoscopy, volume rendering.

1 INTRODUCTION

Minimally invasive endoscopic procedures are gaining importance, since they lead to a reduced trauma and hospitalization duration. Endoscopic interventions are challenging for the surgeon due to the indirect and limited view and the difficult eye-hand coordination. The surgeon has to infer the complex spatial relations solely based on a view at a two-dimensional screen and cannot use palpation for orientation and navigation. Therefore, virtual endoscopy (VE) systems have been developed in order to simulate these interventions either for training purposes or for planning an actual intervention based on the CT or MRI data of the patient. Such VE systems are already used in different areas, e.g., in pituitary gland surgery and other neuroendoscopic interventions [15, 2] or for bronchoscopy [1]. In general, surgery planning is considerably more difficult to support than training, since the large variety of data from the clinical routine has to be processed efficiently and robustly, instead of using only one carefully prepared dataset.

Functional endoscopic sinus surgery (FESS) is a minimally invasive intervention, which is applied in order to treat patients with extended mucosal swellings in the paranasal sinuses [13]. This procedure represents a huge advantage for the patient, taking into account that open surgery would lead to severe cicatrices. These interventions are frequently performed. In selected cases, they are risky and difficult to perform. Based on anatomic variants or the extent of mucosal swelling it might be necessary to operate, e.g., close to the optical nerve or the skullbase, which must not be hurt. Relapse surgery is also difficult, since important anatomic landmarks are missing due to a prior intervention. In this paper, we present an endoscopy system targeted at the special requirements of FESS interventions.

The presented system called *Sinus Endoscopy* is based on CT datasets which are routinely acquired for diagnosis and preoperative

planning. A major requirement for surgical planning is a high degree of realism in order to achieve a sufficient degree of similarity between preoperative visualizations and the intraoperative view with respect to the anatomy and the properties of the endoscopy system (e.g., surface wetness). Furthermore, the surgeon should be able to explore the data efficiently without tedious manual processing. The design of the *Sinus Endoscopy* system is based on these requirements, and thus, focused on an optimized workflow. A major difficulty is to process datasets with swollen and path structures, often with nearly closed airways. Furthermore, we provide the possibility to draw sketches in the virtual endoscopy to mark hard-to-see structures. The system is optimized to run on off-the-shelf hardware or notebooks, but also provides higher visual quality on modern and more powerful computers. All volume rendering steps are directly implemented on the GPU to reduce the CPU workload. The development from an initially flexible but complex prototyping system to an application with carefully reduced functions with default values and templates will also be described.

2 PREVIOUS AND RELATED WORK

The diagnostic value of VE for the diagnosis of sinus diseases was investigated in [17, 3, 8]. The representation of 3D information is realized by volume or surface rendering. The 3D models are used to generate images and videos of different anatomical structures from the view of an endoscope for diagnostic purposes. Comparisons between a virtual and real endoscopy point out that the anatomical structures and variations of the sinuses are recognizable in the 3D model (cf. [8]).

Different prototypes have been developed for the training and simulation of FESS (e.g., [21, 16, 14]). Besides a realistic representation of the structures, the design is focused on realism and faithfulness. Pössneck et al. [16], for example, use a training system adapted to endoscopic sinus interventions for the simulation. A segmentation and modelling software is used to create a deformable polygonal 3D model. Such training systems are too expensive and require too much preprocessing time for the regular planning in clinical settings (cf., [14]). Furthermore, an exact simulation, e.g., of tissue deformation, is not essential for the task of intervention planning. The STEPS system [15] supports the planning of endoscopic tumor operations to the brain using sinus cavities as access. It also provides a collision detection with force feedback navigation, which is useful for the orientation of the surgeons. However, in the STEPS system the entry to the brain only uses the healthy sinuses in a direct way, where no swollen or

-
- Arno Krüger, Christoph Kubisch, and Bernhard Preim are with the Department of Simulation and Graphics, Otto-von-Guericke-University of Magdeburg, E-mail: krueger@isg.cs.uni-magdeburg.de.
 - Gero Strauß is with the ENT Department, University Hospital of Leipzig, E-mail: Gero.Strauss@medizin.uni-leipzig.de.

Manuscript received 31 March 2008; accepted 1 August 2008; posted online 19 October 2008; mailed on 13 October 2008.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

pathologic structures occur. A short and preliminary version of the work, presented in this paper, was published at a national workshop [11]. Here, we extend this work with a substantial evaluation and a detailed description of the rendering techniques.

3 MEDICAL BACKGROUND

In sinus surgery, minimally invasive FESS techniques are common [10]. With the *Sinus Endoscopy* system we aim at an enhancement of the conventional intervention planning on 2D images with the possibility to perform a VE preoperatively and directly with the clinical data. To achieve this goal, segmentation steps are avoided to cope with the limited time in the clinical routine. ENT surgeons stated that the additional planning time for VE should not exceed two minutes.

3.1 Endoscopic Sinus Surgery

FESS is a frequently accomplished intervention, but very risky in selected cases, e.g., if important anatomical landmarks were removed during earlier operations. Unusual anatomical features (stationary skullbase) as well as closeness to risk structures (e.g., N. opticus or A. carotis interna) can further increase the risk level. A similar spatial representation with VE can support the intraoperative orientation as a preparation step and help finding structural features that cannot be seen easily on the usual 2D CT slices. A minimization of the risks mentioned above is already possible through the 3D view of a VE. The quality of planning would even be enhanced with a segmentation of the mentioned risk structures and a combined visualization. However, this adds additional time to the planning process which is often not available. Figure 1 shows the location of the sinuses, which are the intervention field during FESS.

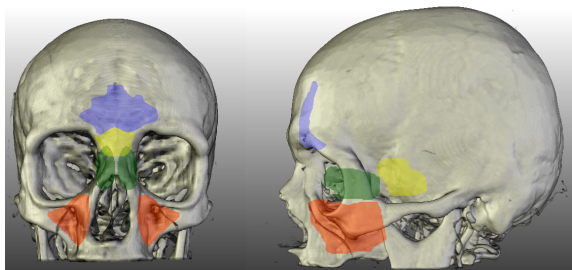


Fig. 1. Anatomical structure of the sinuses. Red: Sinus maxillaris, Yellow: Sinus sphenoidalis, Blue: Sinus frontalis, Green: Sinus ethmoidalis.

3.2 CT Datasets

For surgical planning, CT data are acquired, because they deliver a good representation of the bony anatomy and the mucous membrane changes, and they enable the radiologist to assess the severity and the localization of chronically phlogistic sinus diseases. Nose polyps and possible sequelae can also be located exactly. A spiral CT with an axial 1 mm slice distance is currently the optimal resolution. In clinical datasets also larger slice distances up to 3 mm occur. Due to the filigree sinus structures, higher resolutions are preferable to provide a faithful anatomical representation. Datasets often show closed airways by mucous, polyps or other pathologic variations, depending on the diagnostic finding. This is a big challenge for a VE system and has not been tackled so far. Another problem is the variety of the CT-DICOM data from the different scanners. Besides this, the patient position changes or sheared data can occur and have to be converted during the loading.

4 VOLUME RENDERING

Next to the interactive camera controls, the key element of a VE system is the visualization of the volume data. Because analyzing slice images needs a trained eye to mentally reconstruct the 3D structure, volume rendering allows a faster shape recognition.

4.1 Requirements

Surgeons need a realistic visualization of the nasal anatomy to recognize the individual shape of the patient during the pre-operative planning. One goal of *Sinus Endoscopy* was to enable surgeons to quickly load and browse the data (to fulfill the estimated planning time <2 minutes). For the records of the patient's case it is useful to archive the data for later viewing to support comparisons of volume datasets or documentation with screenshots or animations. The presentation of the diagnosis to the patient is another reason for a realistic, but slightly stylized imagery being used. The rendering system of *Sinus Endoscopy* takes the following aspects into account:

Realistic visualization: Illumination and depth cues via texturing of the tissue surrounding the sinuses enhance the shape recognition. Other shading effects, such as wetness and the secretion level inside the nose, should be graded to present structural information in a way that is well known from real endoscopy. An added lens distortion simulation also effects the characteristics of the physical reference. The renderer should enable the integration of modes with greater realism, or the opposite, stronger stylization.

Usability: We avoid image segmentation, since it takes too long for regular surgical planning (cf. [16, 14]). For documentation and planning purposes, the possibility to draw sketches on the tissue during the endoscopy should be integrated. Usability also means that the system can easily be used on consumer class hardware.

Customizations: The CT Hounsfield units might slightly vary depending on the case, thus the system should allow interactive changes of the levels of density that serve as thresholds to solid tissue (app. -230 HU), secretion and air. This gives enough flexibility for an instantly useable visualization that can yet be reproduced by defining default values. It is also useful to cope with small passages due to swollen or pathology structures, but if the cavities are totally closed we cannot fly through them. Users might also prefer different parameters of the rendering (e.g., wetness level or secretion color) and adapt parameters to enhance, e.g., the brightness of the monitor. Hence, besides the standard values, the renderer should have the flexibility in these shading parameters.

Currently, no system for VE fulfills all criteria simultaneously, especially not the fast and easy use. The nasal sinuses are an area with both large hollow shapes, and fine labyrinths of channel systems. Furthermore, density values are caused by diseases affecting the secretion transport. This leads to a challenging rendering problem. The STEPS system is the only (commercially) available system for ENT and neurosurgery planning, but comes as a plug-in for a radiological workstation. This comes along with an overhead of GUI elements, which is not mentioned in the very good STEPS interface. Still, the rendering here is optimized for look behind the wall applications and no natural look of the tissue is possible. STEPS, in contrast to *Sinus Endoscopy* allows, to interact with the volume data, e.g., to drill holes. This is a useful function and should be added to *Sinus Endoscopy* later.

4.2 Renderer Description

The *Sinus Endoscopy* system makes use of the latest technology to generate perspective images at realtime frame rates as long as datasets fit directly into the video memory. This limitation has not been an issue in clinical routine (usually <200 slices with 512×512 pixel). The image shading was tailored towards realism and is realized using ray-casting. At first the secretion layer is detected and its density is accumulated. Finally the ray stops when finding tissue, which is represented as textured surface. To give further shape cues, a specular wetness effect is applied.

Ray casting is employed to combine volume rendering for secretion accumulation with isosurface rendering for the mucosa. With advancements in pixel shading capabilities the current generation hardware allows ray traversal within a single pass. Stegmaier et al. [20] presented

a framework that uses such single pass ray traversal for volume rendering. Thus, it does not require complex setups anymore, which used multiple drawcalls to benefit from speed enhancing principles, such as early ray termination seen in [12]. As ray-casting classifies pixels at runtime, the user can interactively change thresholds for secretion and tissue classification.

Isosurface extraction is typically performed via algorithms such as Marching Cubes. We decided against generating triangular meshes, since a high resolution would be required to faithfully represent the fine inner anatomy of the sinuses, resulting in a large amount of triangles. The method was rated less ideal for this application, because real-time marching cubes seemed harder to achieve on older generation hardware (cf. [4]). Furthermore, we would have less flexibility in shading and that the secretion cannot be accumulated along view-rays.

Ray casting of the surface does not necessarily need a full triangulation of the volume. With acceleration structures, empty regions are skipped. However, when rendered via CPU, most computers still do not have enough processing power. Recent advancements in the programmability of graphics hardware made it possible to render such surfaces with off-the-shelf graphics hardware. *Sinus Endoscopy* uses such a technique for tissue surface finding with GPU ray casting (described in detail in Sec. 4.3, cf. also Fig. 3).

Classic direct volume rendering via simple blending is also not appropriate for typical situations in VE. The performance degrades quickly when the user is inside the volume, due to high fillrate costs.

Illumination and final coloring in *Sinus Endoscopy* is performed via deferred shading (cf. [7]). The main volume pass gives results, such as the depth of hard tissue in viewspace, density values or other attributes for each pixel of the final image. The composition and shading is performed in another pass to get the final image. This allows a flexible customization of the shading, and also the use of enhances attribute buffers in between. While the technique is not optimized for large datasets, or for renderings when the camera is outside the volume like in [18], for this sinus surgery planning the results are visually appealing at very high frame rates on standard hardware.

4.3 Quality and Speed Enhancement

To achieve high quality on current hardware while still allowing the interactive exploration of the dataset, GPU ray casting techniques have been applied. An important acceleration strategy that can also be realized on the GPU is first-hit early ray termination [15]. Once the isosurface has been hit or the ray left the volume, the search along the ray is stopped. For VE applications this is reasonable because the surface shape is the most important information to visualize. A transparent look through the whole dataset, as presented in other publications (e.g., [18]), is not the goal. This assumption results in much lower requirements in the necessary GPU calculation power. The renderer uses ray casting with varying step distances (planar slices) for the speed optimization and easier compositing with regular depthbuffer, which encodes planar distances. For empty space skipping, only a simple method was employed (see Figure 2), since the user would most of the time be within the volume and close to surfaces.

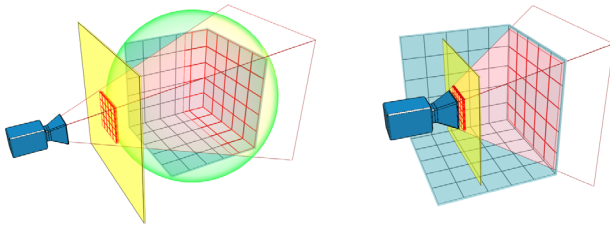


Fig. 2. Primitive empty space skipping when outside volume, starting rays at tangential plane (left), otherwise near plane (right) is used.

When inside the volume, no empty-space skipping is done and rays are started directly from the camera’s near plane. However, the system

should still allow the user to move outside the volume, where empty space-skipping is more important to keep interactive framerates. In this case, instead of the near-plane, a plane tangential to the bounding volume is computed by the CPU and used as starting grid for the view-rays. End points of the rays are always generated by rendering the volume hull as box with faces facing inside.

Since the search for the isosurface transition runs at fixed stepsize, the final hitpoint might not be accurate enough, if the last sampling point was close to the surface. This yields noticeable visual artifacts. Decreasing the step size also means more samples in “empty areas”, and thus hinders the performance. Sigg et al. [19] used a binary search refining step once a sample inside a volume was taken. A fixed amount of iterations greatly enhances the quality and does not strongly affect the performance. The secretion level can either be accumulated per sample along the ray or approximated by finding the surface that marks the “upper” bound of the viscous material, using the same way as the solid surface. The basic procedure for the latter method is presented in Figure 3. Each ray outputs a solid and secretion distance to the starting plane and the last position in volume. Additional data, such as secretion density or masks if a hit was found at all, are also possible to be encoded into texture outputs.

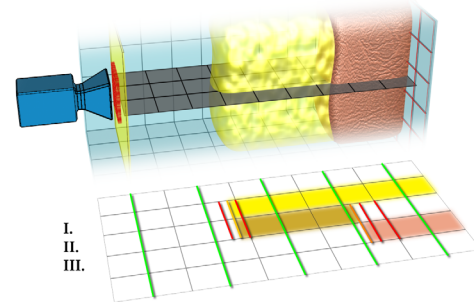


Fig. 3. Single ray sampled at fixed steps (green). The result is refined (red) to give more precise results. Secretion density (II) is assumed constant between first secretion hit (I) and first solid hit (III). Alternatively for high-end hardware density can be accurately accumulated at a lower step size. Solid hit terminates the process.

To further reduce parallel plane sampling artifacts, interleaved sampling was successfully used on GPUs by Scharlach et al. [18]. The rays are randomly offset a bit to prevent neighboring pixels to have the same starting point. Especially when secretion accumulation was used, this improved the visual quality. Figure 4 shows disabled and enabled quality enhancing methods. For maximum efficiency in texture fetches, 8-bit single channel textures were used for the volume data. The precision loss of the original 12-bit density values is reduced on load. Since the minimum and maximum density values of interest are specified before, only 1 or 2 bits are lost. The benefit is a small memory footprint and more texels are fitting into the texture cache. However, the system also allows the loading of 16-bit precision volumes. The filtering of the 3D textures is improved by using custom spline filtering as presented by [7]. This is only useful for still rendering, as it severely affects the interactivity.

4.4 Texturing and Composition

The output values of the ray-traversal pass are employed to determine the final shading. The last position is used to look up a detail texture. The texture is projected along each major axis, and via blending combined through the surface normal. The listing in Figure 5 shows the used blending formula, which was taken from [6]. The normal is either generated via gradient estimation using the hit positions of surrounding pixels in the 2D output buffer or with central difference sampling neighbors inside the volume. The latter yields the most appealing results. Figure 6 shows the same scene with different detail textures, and indicates that it improves the shape recognition. Normal, depth and a dedicated buffer to represent the secretion are smoothed

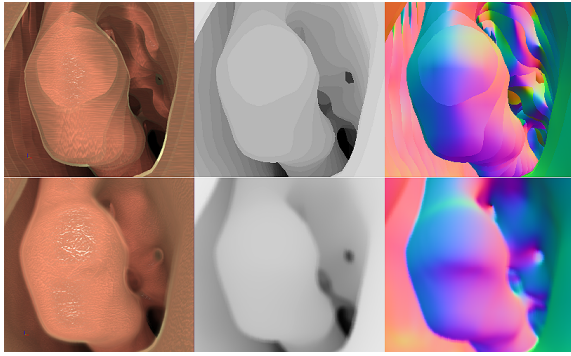


Fig. 4. Top row shows no refining being performed, whilst bottom uses refinement with binary search and interleaved sampling. Additionally, the outputs were smoothed in image space.

in image space to remove noise or other artifacts. The smoothing is performed anisotropically and takes the depth difference of neighboring pixels into account, which can also produce bent normals, e.g., for stronger silhouettes.

Illumination is performed using standard Phong lighting or image-based lighting, using 2D textures as presented in [22]. An important depth cue for shading is the attenuation of light, causing more distant surface points to be darkened. Specularity is currently linked to darker pixels of the detail texture and is the key part of the wetness effect (cf. Fig. 7). The wetness level is determined similar to specular with a power function of the $-$ view direction and surface normal $-$ angle cosine using low power values. This level darkens areas accordingly before the actual specularity is added. Both wetness and tri-planar texturing have been implemented similar to [6]. The secretion level also influences this process, raising wetness depending on the density of the accumulated secretion and the distance of its first layer. The primary effect of secretion is to fade out the tissue below, similar to fog by linear blending from tissue- to fogcolor (see Fig. 8).

More complex texturing methods can be applied to use dedicated specular maps and colored detail textures. The use of monochrome detail textures has the benefit of packing each tri-planar contribution in a single color channel to be output at the end of the main volume pass. We reconstruct the 3D positions after smoothing, since directly using original hitpoints was visually inferior. Detail-texture sampling is therefore performed in the final composition. Both performance and quality were positively affected by this, as all inputs for reconstruction do not depend on volume's pass branching issues anymore. Therefore mipmapped texture access, which depends on pixel neighbors, was less distorted and yielded better results. Using the final world positions of the tissue surface, additional textures can be projected using regular texture projection as presented in [5]. It allows a frame-consistent drawing on the tissue surface, as long as the transfer functions are not changed. Snapshots of the projected color image as well as the corresponding depth buffer are used to ensure that the projection only affects the pixels seen in the original camera view.

Drawing Facilities. According to the requirements surgeons can create markings on the surface to specify the intended intraoperative procedure (see Fig. 9). *Sinus Endoscopy* provides two standard tools, a semi-transparent brush and an opaque and much smaller pencil. After drawing, the overlaid sketch can be fixed to the surface as a texture. The drawn color and the tissue depth values are transferred to a texture atlas upon user interaction. The atlas can store several projection slides. The current view and projection matrices are stored, and later used to create coordinates for the texture lookups by classic projective texturing. Additionally, for each pixel a depth comparison similar to shadow-mapping is carried out prior to apply the color as decal. This ensures that the tissue position in space matches the one that was stored before. This way, the drawing remains on the virtual walls, even by moving the camera, or disappears if the tissue is removed by thresholds. Up to four texture projectors can be applied, but usually one will

```
// input = normal, pos (world space)
// output = detail

// Smooth weights according to normal
blend_weights= abs(normal.xyz) -0.2;
blend_weights*= 7;
blend_weights= pow(blend_weights, 3);
blend_weights= max((0,0,0), blend_weights);

// Factors must sum to 1.0
blend_weights/= dot(blend_weights, 1);

// Sample 2D textures with scaled positions
// for texture repetition.
// One color for each planar projection direction.
colx = tex2D(texdetail, (pos.yz)*texscale.x );
coly = tex2D(texdetail, (pos.xz)*texscale.y );
colz = tex2D(texdetail, (pos.xy)*texscale.z );

// final color is mix of each projection
detail = (colx*blend_weights.x) +
         (coly*blend_weights.y) +
         (colz*blend_weights.z);
```

Fig. 5. Pseudocode block of the weight calculation for the blending of the tri-planar texture mappings.

be enough for planning or explaining structures to patients (Sec. 5.1).

Integration of Polygonal Data. The buffers of previously rendered polygonal data also serve as input for the composition. Polygonal meshes are primarily used for integrating segmented structures, e.g., to show vessels behind the tissue. By using the depth buffer of previously drawn meshes the combination of both volume and meshes can take overlaps into account. This results in frame-coherent intersections between the two different renderings. The secretion level can also be reduced, if the mesh depth is closer than the tissue depth. Furthermore, stippling techniques can be applied to provide a “look behind the wall” functionality, and to render meshes that are otherwise occluded. A final color overlay is performed when the tissue is being clipped, or if one is inside a polygonal mesh. The latter is achieved by using the stencil buffer and marking backface pixels alpha values, and the former by very close depth values of the tissue surface. Further effects can be easily incorporated, since deferred shading is very flexible and only has image space complexity.

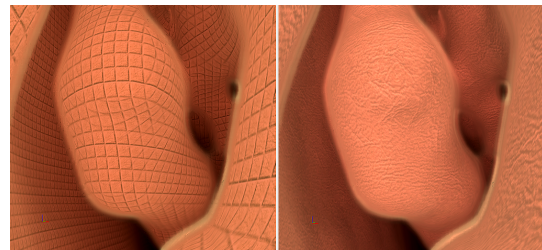


Fig. 6. Different textures were applied. Overlapping distortion artifacts resulting from blending, as seen left, are hardly visible with organic textures used right. However, grid textures yield stronger shape clues.

5 IMPLEMENTATION OF “SINUS ENDOSCOPY”

Sinus Endoscopy was developed using the *Luxinia* 3D engine as platform (<http://www.luxinia.de>). The engine itself is programmed in C for portability purposes and for speed critical tasks. Application coding is performed with the scripting language *Lua* developed by the Pontifical Catholic University of Rio de Janeiro, Brazil (<http://www.lua.org>). *Lua* allows an object-oriented programming, and being a high-level language it shortens the development time. The *Luxinia* API offers a rich functionality to aid the rapid-prototyping of

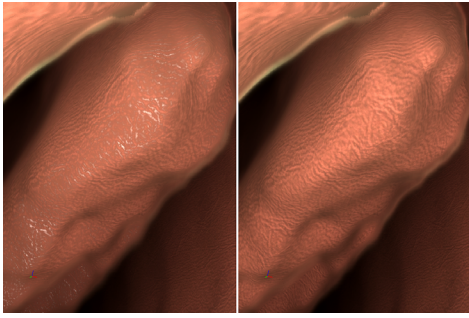


Fig. 7. Varying shading parameters of wetness effect. Left: natural amount of wetness. Right: no wetness.

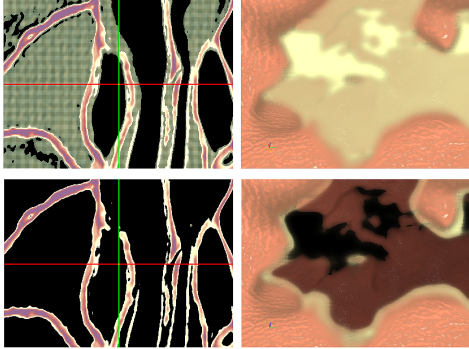


Fig. 8. The interactive transfer function allows revealing structures and mucous obstruction without the need of any pre-processing.

3D games and applications. OpenGL is the rendering backend of *Luxinia*, and *Cg* was used for programming the graphics hardware. With the fast scripted application prototyping it was possible to implement various VR techniques to perform speed measurements and to build and refine prototypes with low programming effort.

5.1 Application Structure

The *Sinus Endoscopy* system consists of five main modules. They communicate with each other via a callback mechanism depending on user interaction, as shown in Figure 10.

View: There are three orthographic views along each major axis depicting the slice planes in which the perspective camera resides. The perspective view uses the active method for volume rendering. Volume rendering is performed to a 512×512 render target, which is scaled to the final resolution. In the orthographic views the density values of the volume are colored via a hue gradient texture and stippled if secretion occurred inside.

Volume Data: It contains the 3D texture, which stores all density values, and the scaling dimensions of the final volume box. A copy of the data is stored in the application memory for collision detection. The CT dataset is loaded and commonly converted to 8-bit alpha textures (recall Sec. 4.3). If the graphics hardware does not support non-power-of-2 textures, the final volume position in 3D space is taken into account. This is necessary to make the camera position consistent on various hardware.

Rendering Method: The rendering setup for the perspective view is stored in this class. It loads all shaders, manages their parameters, and creates the necessary render targets. Methods can be exchanged and reloaded on the fly, allowing a fast scaling of the visual quality, and be handled by the event processing module.

Animation: This module provides an interpolated key frame animation between camera states. The position of the camera is interpolated using Catmull-Rom-Splines and rotation via quaternion

spherical interpolation. In this paper we do not discuss the animation possibilities in detail.

Painter: The paint module allows to overlay a texture that can be painted on with the mouse (Brush events). If the current rendering method supports it, the painted area above the perspective view is projectable on the tissue (cf. Fig. 9).

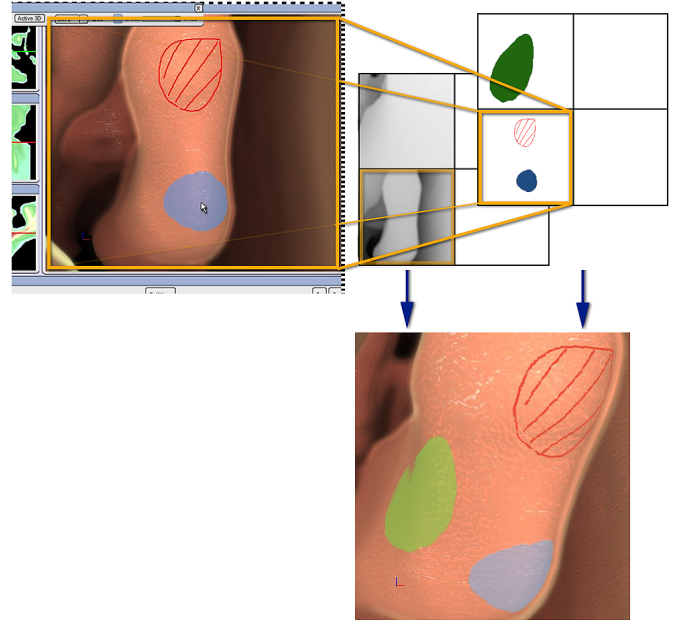


Fig. 9. The user can draw on the tissue surface for patient education or planning. A snapshot of the drawn image is stored into a projection atlas texture, along with the tissue depth. By transforming the tissue position with the snapshot's view-projection matrix texture coordinates are generated for the lookup. Frame coherency for texture projection is reached by depth-comparison on pixel level to make sure that it was the same position in space as those stored in the user-drawn textures.

5.2 Adaptation to Different Hardware

Since *Sinus Endoscopy* is an application for clinical use, it should be useable without requiring the latest high-end video cards. Most recent hardware as well as early generation cards, which support the minimum requirements, should be able to attain interactive frame rates. There are two principle setups: Either the volume pass and final image output are performed in the same pass and directly rendered on top of the polygonal render buffer, or multiple passes are used and a final composition is the final step. The second method allows the highest flexibility and quality as attribute buffers, holding depths or normals and can be further smoothed to reduce artifacts. However, this results in certain graphic driver-related overheads and possibly more computations, which decrease the performance. Figure 11 shows the result of the cheapest method that is directly rendered, and one with all composition effects that is realized with a dedicated composition pass. To support the use of low-end systems, the resolution of the main render target can be quartered to 256×256 . The resulting loss of sharpness and precision is not too adverse, since the typical clinical volume resolution is not very high, and the anatomy is rather organically smooth. When no camera movement is performed, single still frames can be rendered with higher quality, as the system only updates the endoscopic rendering view when changes were made. This allows a low hardware consumption when the user performs no significant actions.

5.3 Performance Measurements

Tests have been carried out on four hardware configurations (cf. Figure 12). Polygonal rendering was not used in any test. The interactiv-

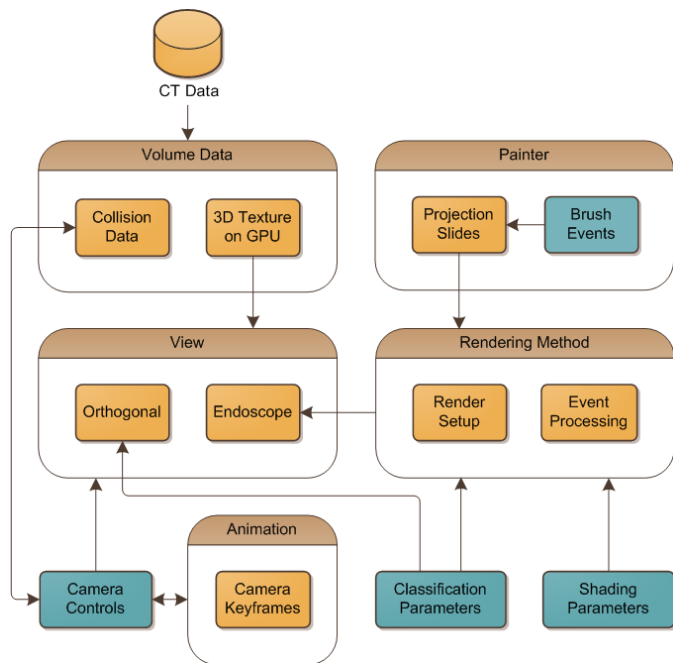


Fig. 10. Overview of software modules (orange) and basic interaction messaging system. The input modules used in the GUI on the bottom of the diagram are colored in blue.

ity mostly varies on how much empty-space has to be skipped. The timings presented are averages in a flight into the nose, starting inside the front tip, with consistent values on maximum view distance and sample step size. The latter two directly affect the performance and quality, as they result into more sampling being done. Raising the total amount of sampling yields asymptotic performance loss. Early-ray termination helps to reduce sampling, hence frames with lots of tissue up-close are much faster than those with large cavities. However, due to the nature of the anatomy, there is normally a similar bias of closer and more distant hits.

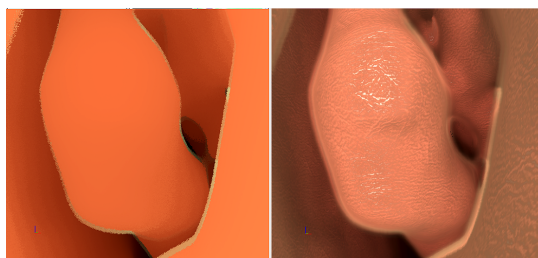


Fig. 11. Visual comparison of the rendering results. Left: For low-end hardware. Right: Full featured image for more powerful GPU's.

Volume sizes did not significantly affect the performance, as hardware efficiently takes care of sampling large 8-bit single channel volume textures. We employ a full dataset ($512 \times 512 \times 128$) and a cropped version ($256 \times 256 \times 64$). The basic direct method and a more complex anisotropic filtered composition method were compared using the two different render target sizes. Only the complex method does shading with textures and can apply projected user-drawn labels. However, the labeling texture activation had no significant impacts on speed. In contrast the basic method only darkens based on the distance. Both techniques refine hitpoints and interleave rays, and can show secretion as fog effect.

When the lower resolution was used, the slowest system also achieved more than 20 frames per second (fps) in the composition techniques. The measured value for the quartered resolution on the

GeForce 8800 system (last sample on purple curve) is CPU-limited and would be higher, if identical CPU setups were used (even though more than 300 fps are only of theoretical interest). On a standard PC with a modern GPU always more than 30 fps are possible on highest quality settings. This comes along with a typical CPU utilization of 10-15%. This value does not depend on the size of the dataset, as long as the data fits into the GPU memory.

Clinical data of the head always fits into standard graphics cards with 512 MBytes of RAM. The biggest high-end dataset has some 180 slices and 90 MBytes. Typical data have 35-140 slices resp. 17-80 MBytes. Based on the past development we expect that also a GPU with 256 MBytes of RAM can cope with datasets in the next years, even if the head of the patient will be scanned with slice distances around 0.5 mm. 512 MB dedicated RAM for the GPU will be enough for real time rendering 0.2 mm (more than 1000 slices for the head!). Due to the growing radiation of the patient during scans with higher resolution such scans would also be avoided in future medial applications.

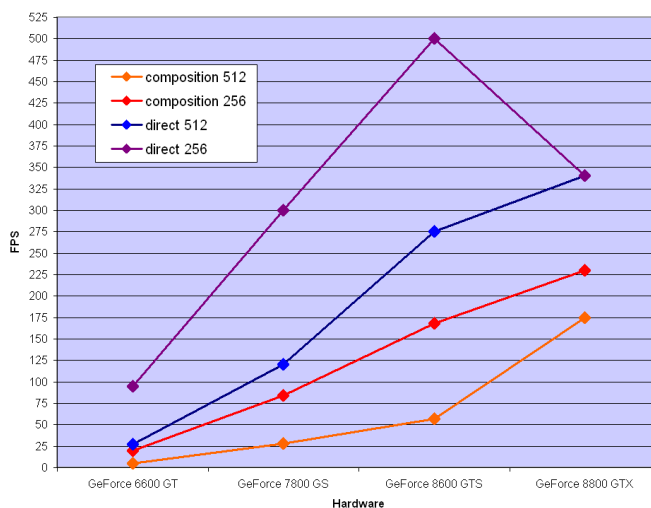


Fig. 12. The techniques scale well with advancements made in graphics hardware and can achieve interactive frame rates for all systems.

5.4 Interaction and User Interface Design

The user can interact with the application in various ways. The orthographic views allow panning and zooming the volume data, while the perspective view handles the camera rotation and movement. The camera movement is limited by a simple collision system that blocks the camera from penetrating the tissue by rejecting camera moves to positions that lie in the solid volume. This collision test is evaluated by the CPU and a copy of the volume data residing in RAM. Changes to the camera are also automatically registered with the animation system for creating or changing key frames. Volume rendering is mostly controlled by defining the threshold from solid and air, and defining the secretion classification. Next to those global parameters, each method implements a series of parameters that allow to tweak the final image. Shading parameters, such as light intensity, can be easily manipulated using sliders or color menus, showing their influences immediately. Polygonal meshes can be loaded into the perspective view and their transparency can be changed. The application has been developed for an efficient workflow, especially to start immediately and is very lean on the system resources.

The first prototype had many different switches to directly test clinical datasets on standard hardware with these VR renderings. A second prototype was built to refine different GUI variations and parameter settings for the rendering as well as usability aspects (see Fig. 13).

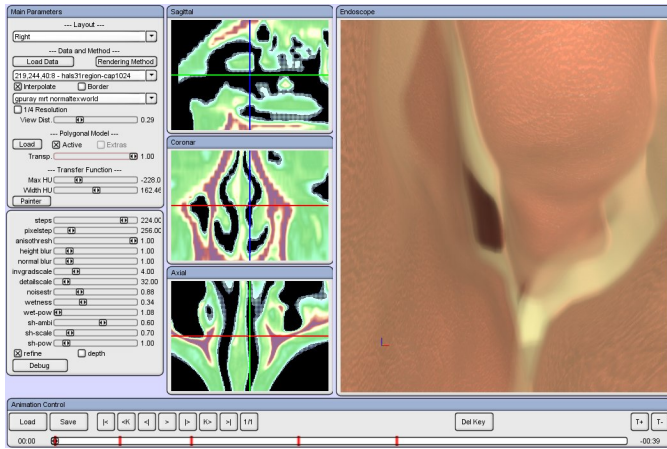


Fig. 13. Screenshot of the prototype's interface. Slider elements allow fast changes to values, yet numerical inputs are also allowed for precision. Animation controls are located at the bottom, while main parameters are at the left sidebar. In the upper section there are, e.g., the iso value and secretion level while the lower box contains parameters who influence the visual effects of the GPU shader.

5.5 Clinical Evaluation

To prepare the clinical study, we carefully developed a robust prototype for use in the routine. As described in Sec. 5, three prototypes were built - each with major improvements in usability and functionality.

Initial Feedback. To investigate the usability, speed and visual quality of the early *Sinus Endoscopy* prototype, we observed an ENT surgeon using the system for planning an actual intervention. He appreciated the similarity to intraoperative views and the usability of the application. The surgeon optimized visualization parameters like transfer function setting, texturing attributes and the visibility of secretion. The range of these parameters was already shortened to enable an optimized fine tuning. A representation of the current camera viewpoint is also useful for orientation and was already added in the prototype for clinical evaluation. This last redesign lead to a reduced GUI design with the necessary sliders and buttons for a clinical study. Here, the focus was to finally achieve meaningful default rendering parameters. Figure 14 shows this prototype - most of the few sliders left will later be replaced by buttons with the identified standard values. Only the sliders for tissue threshold, secretion threshold and for the illumination are most important. The differences between datasets and different monitor models make these adjustments necessary.

Demonstration at ENT Workshop. During a workshop with European ENT surgeons the *Sinus Endoscopy* system was demonstrated with cadaver datasets. The cadavers could also be operated and investigated parallel with real endoscopes and instruments. The surgeons considered the clear visualization of the Rec. frontalis or the Concha nasalis medialis as essential.

Application for Middle Ear Surgery. During the workshop, another application for *Sinus Endoscopy* was presented: the VE of the ear. Middle ear surgery poses a variety of requirements which are very similar to sinus surgery. Again, a complex bony anatomy, consisting of anatomical landmarks, has to be represented. Wetness of tissue and secretion are again challenging aspects. In order to evaluate whether *Sinus Endoscopy* is also applicable for this application area, we generated virtual endoscopy views for two cases. Therefore, the two HR-CT datasets were loaded, and besides the reduction of the secretion value nearly to zero, no parameter adjustments had to be done (see Fig. 15). The feedback of the surgeons was also positive with respect to the recognizability of structures. With a real endoscopy it would not be possible to prepare an operation, due to the invasive nature of this task.

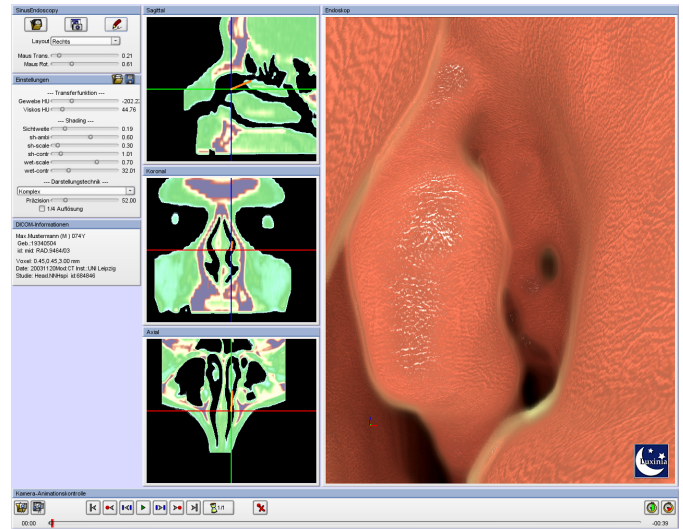


Fig. 14. Strongly reduced user interface for parameter evaluation. On the left side a patient information box was added.

Clinical Evaluation. Based on the feedback obtained so far, we further improved the *Sinus Endoscopy* system (e.g., reduced sliders and optimized default values for thresholds and tissue effects). A study at two clinical sites was accomplished to achieve final parameter values, to test the usability and robustness in general, and to compare the intraoperative views with the preoperative planning. In both cases a standard PC (with low end graphics card: nVidia GeForce 8600 GS, 512MB) was used with an existing big wall-mounted plasma display as a secondary screen. This display leads to an optimized use in terms of visibility and patient education. For these relatively dark displays the mentioned sliders for the rendering parameters were used to optimize color and brightness.

So far, the system was used for preoperative planning of sinus surgery in 102 cases. There is a trend to a better understanding of the patient individual anatomy and of the planned intervention compared to the traditional planning with 2D slices. The investigation of the Rec. Frontalis benefits from the system due to the very individual anatomy of this structure. The similarity between the preoperative Virtual Endoscopy and the real anatomy in the intervention was assessed on a 5 point Likert scale (1 represents identity and 5 no correspondence). An average score of 2.21 confirms a good correspondence.

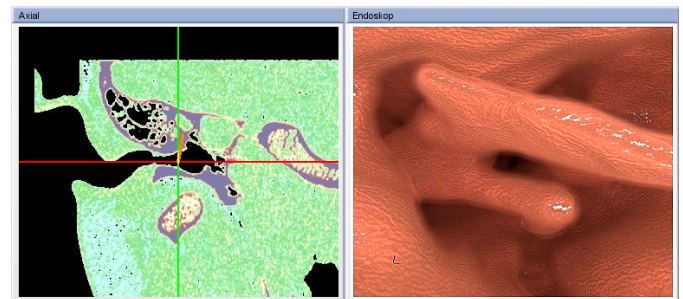


Fig. 15. The system can also be used to explore other hollow organs such as the inner ear that could not be accessed with real endoscopes.

Furthermore, it turned out that the *Sinus Endoscopy* system can be used quickly (<2min) and the remaining rendering parameters were judged as useful. Real endoscopic screens and devices also have various controllers to enhance the video quality. Hence, finding optimized parameters is a common task and user individual preferences are possible. The variation of the parameter secretion level is useful for less experienced surgeons, and patients. The nose functions can be sim-

ulated and convey the understanding why a small secretion level can already close the main cavity of the nose.

It could also be observed that important landmarks, e.g., the Concha nasalis medialis, are clearly visible in the VE. Furthermore, additional features of the patient anatomy are visible using the VE system together with the orthogonal 2D views. The drawing possibilities are judged as useful in general, but are not used in routine. Especially for the documentation and marking of hard-to-see structures or explanations to a patient they could be useful in future. Furthermore, the drawings for surgical applications must be later usable during intervention, e.g., combined with navigated control.

The natural look of the renderings was evaluated as a key feature of the system. Only with such a rendering the patient can be convinced – more abstract visualizations cannot accomplish this. To avoid misinterpretations the look is still artificial enough and the surgeons stated that it is definitely not a real endoscopic image and cannot be overrated.

5.6 Consequences of the Clinical Evaluation

During the evaluation, the simple collision detection sometimes lead the camera too deep into dense structures. The storage of the drawings in connection to the case was also a requested feature. This documentation feature will result in a redesign of the data handling in *Sinus Endoscopy*. Surgeons stated that with the drawing feature also the workspace of a computer controlled shaver could be defined (see [9]). The data exchange with the control system will be the next step in this application.

Another requested feature was to deform the tissue to see, e.g., the bottom of the tongue. A deformation functionality will also enable to push structures virtually slightly away. This supplements the adjustments of the threshold to reach even more nearly closed airways.

6 CONCLUSION AND FUTURE WORK

We presented a virtual endoscopy application that integrates state-of-the-art rendering techniques and integrates realistic textures to inform the patient. The system is dedicated to plan Functional Endoscopic Sinus Surgeries (FESS) and is in clinical use since more than 100 cases so far. Using GPU-based volume rendering greatly decreases the CPU workload and enables existing computers in hospitals to use the *Sinus Endoscopy*. The estimated time of approx. 2 min for additional planning was reached by the possibility of loading CT datasets and visualization with direct volume rendering without any further steps. With the *Sinus Endoscopy* system it is possible to blend in segmented structures in the endoscopic view to provide a look “behind-the-wall”. This feature is not used very often, due to the limited preparation and planning time per case. However, in selected cases it is very useful.

The rendering techniques presented here can be used efficiently with a variety of GPUs. With this approach, significantly more frames per second (min. 20 fps) than necessary for interactive framerates (10–15 fps) can be achieved, even on GPUs from the previous generation (GeForce 7). We intend to provide a feature to automatically adapt the quality to the currently available graphics hardware in the next version, thus, removing the need to manually specify these parameters. Another work in progress is to directly use the surface-painting function for standardized case documentation purposes. A clinical study with the system has already started to compare the preoperative planning with the intraoperative situation. In this first evaluation with approx. 40 cases the visualization parameters are a second major evaluation criterion.

The *Sinus Endoscopy* system has also been used for virtual endoscopy of the inner ear and could render those datasets instantly with a high similarity to the anatomy. The methods presented here can be probably applied to other areas, e.g., virtual bronchoscopy. Adapting the textures is the major issue.

ACKNOWLEDGEMENTS

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) (Priority Programme 1124, PR 660/3-1, PR 660/3-2). Special

thanks go to Ragnar Bade for his good ideas and substantial support on visualization issues.

REFERENCES

- [1] D. Bartz, D. Mayer, J. Fischer, S. Ley, A. del Rio, S. Thust, C. Heusse, H. Kauczor, and W. Strasser. Hybrid Segmentation and Exploration of the Human Lungs. In *Proc. of IEEE Visualization*, pages 177–184, 2003.
- [2] D. Bartz, W. Strasser, O. Gürvi, D. Freudenstein, M., and Skalej. Interactive and Multi-modal Visualization for Neuroendoscopic Interventions. In *Proc. of Symposium on Visualization*, pages 157–164, 2001.
- [3] S. Bisdas, M. Verink, H. P. Burmeister, M. Stieve, and H. Becker. Three-Dimensional Visualization of the Nasal Cavity and Paranasal Sinuses: Clinical Results of a Standardized Approach Using Multislice Helical Computed Tomography. *Journal of Computer Assisted Tomography*, 28(5):661–669, 2004.
- [4] C. Dyken, G. Ziegler, C. Theobalt, and H.-P. Seidel. HistoPyramids in Iso-Surface Extraction. Technical report, Department of Informatics, University of Oslo, August 2007.
- [5] C. Everitt. Projective texture mapping. Technical report, NVIDIA Corporation, 2001.
- [6] R. Geiss and M. Thompson. Cascades. Technical report, NVIDIA Corporation, 2006.
- [7] M. Hadwiger, C. Sigg, H. Scharsach, K. Bühler, and M. Gross. Real-Time Ray-Casting and Advanced Shading of Discrete Isosurfaces. In *Proc. of Eurographics*, pages 303–312, 2005.
- [8] P. Han, W. Pirsig, F. Ilgen, J. Gorich, and R. Sokiranski. Virtual Endoscopy of the Nasal Cavity in Comparison with Fiberoptic Endoscopy. *Europ. Archives of Oto-Rhino-Laryngology*, 257(10):578–583, 2000.
- [9] M. Hofer, G. Strauß, K. Koulechov, M. Fischer, T. Neumuth, C. Trantakis, W. Korb, A. Dietz, and T. Lüth. First clinical trial of the navigated controlled shaver in functional endoscopic sinus surgery. *International Journal of CARS*, 1(1):318–320, 2006.
- [10] L. Klimek, M. Wenzel, and R. Mösges. Computer-Assisted Orbital Surgery. *Ophthalmic Surgery*, 24(6):411–417, 1993.
- [11] A. Krüger, C. Kubisch, I. Richter, G. Strauß, and B. Preim. SinusEndoscopy - Ein skalierbares Visualisierungssystem für die virtuelle Endoskopie. In *Proc. of Bildverarbeitung für die Medizin*, pages 293–297, 2008.
- [12] J. Krüger and R. Westermann. Acceleration Techniques for GPU-based Volume Rendering. In *Proc. of IEEE Visualization*, pages 287–292, 2003.
- [13] G. A. S. Lloyd. *Diagnostic Imaging of the Nose and Paranasal Sinuses*. Springer, 1988.
- [14] A. I. Moral, M. E. Kunkel, K. Tingelhoff, M. Rilk, I. Wagner, K. W. G. Eichhorn, F. Bootz, and F. M. Wahl. 3d endoscopic approach for endonasal sinus surgery. In *Proc. of IEEE Engineering in Medicine and Biology Society, Lyon, France*, pages 4683–4686, 2007.
- [15] A. Neubauer, S. Wolfsberger, M.-T. Forster, L. Mroz, R. Wegenkittl, and K. Bühler. STEPS - An Application for Simulation of Transsphenoidal Endonasal Pituitary Surgery. In *Proc. of IEEE Visualization*, pages 513–520, 2004.
- [16] A. Pössneck, E. Nowatius, C. Trantakis, H. Cakmak, H. Maass, U. Kühnapfel, A. Dietz, and G. Strauß. A Virtual Training System in Endoscopic Sinus Surgery. In *Proc. of CARS*, pages 527–530, 2005.
- [17] P. Rogalla. Virtual Endoscopy of the Nose and Paranasal Sinuses. In P. Rogalla, J. T. van Scheltinga, and B. Hamm, editors, *Virtual Endoscopy and Related 3D Techniques*, pages 17–39. Springer, 2001.
- [18] H. Scharsach, M. Hadwiger, A. Neubauer, S. Wolfsberger, and K. Bühler. Perspective Isosurface and Direct Volume Rendering for Virtual Endoscopy Applications. In *Proc. of Eurovis*, pages 315–322, 2006.
- [19] C. Sigg, M. Hadwiger, M. Gross, and K. Bühler. Real-Time High-Quality Rendering of Isosurfaces. Technical report, VRVis, 2004.
- [20] S. Stegmaier, M. Strengert, T. Klein, and T. Ertl. A Simple and Flexible Volume Rendering Framework for Graphics-Hardware based Raycasting. In *Proc. of Intern. Workshop on Volume Graphics*, pages 187–195, 2005.
- [21] G. Voss, U. Ecke, U. Bockholt, W. K. Müller, and W. Mann. How to Become the “High Score Cyber Surgeon” - Endoscopic Training Using the Nasal Endoscopy Simulator (NES). In *Proc. of CARS*, pages 290–293, 2000.
- [22] H. Winnemöller and S. Bangay. Geometric Approximations Towards Free Specular Comic Shading. In *Proc. of Eurographics*, volume 21, pages 309–316, 2002.