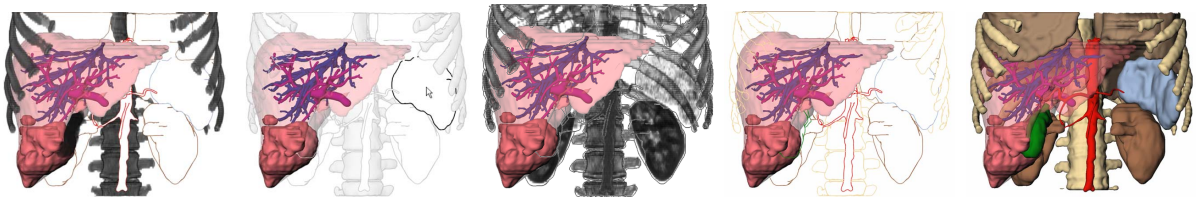
EUROGRAPHICS - IEEE VGTC Symposium on Visualization (2005)
K. W. Brodlie, D. J. Duke, K. I. Joy (Editors)

Combining Silhouettes, Surface, and Volume Rendering for Surgery Education and Planning

Christian Tietjen[1] Tobias Isenberg[1,2] Bernhard Preim[1]

[1] Department of Simulation and Graphics
Otto-von-Guericke University of Magdeburg, Germany
{tietjen|preim}@isg.cs.uni-magdeburg.de

[2] Department of Computer Science
University of Calgary, Canada
isenberg@cpsc.ucalgary.ca



Abstract

We introduce a flexible combination of volume, surface, and line rendering. We employ object-based edge detection because this allows a flexible parametrization of the generated lines. Our techniques were developed mainly for medical applications using segmented patient-individual volume datasets. In addition, we present an evaluation of the generated visualizations with 8 medical professionals and 25 laypersons. Integration of lines in conventional rendering turned out to be appropriate.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Display algorithms; I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

Keywords: Medical visualization, non-photorealistic rendering, hybrid styles, line rendering.

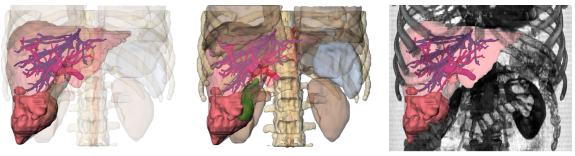
1. Introduction

Direct and indirect (isosurface) volume rendering of CT or MRI datasets dominate in application areas such as diagnosis and surgery planning. Direct volume rendering (DVR) is used for a fast overview, e. g., if complex fractures are involved or when structures should be displayed that do not exhibit (sharp) surfaces. However, using DVR it is often difficult to emphasize objects or their parts. Advanced DVR methods relying on multidimensional transfer functions are better suited for emphasis but they exhibit a huge parameter space that is usually not appropriate for clinical applications.

In contrast to DVR, surface rendering transforms a part of the volume data into a polygonal representation. This is accomplished either as threshold-based isosurface rendering or as surface rendering of segmentation results. The flexibility to adjust surface visualizations is reduced to emphasis with color and transparency. In particular, transparency easily introduces visual clutter (Figure 1). The hybrid combination of surface and volume visualization is often useful: Surface visualization is employed to show anatomic structures which have been segmented in advance, whereas DVR is employed to present anatomic context such as skeletal structures.

Inspired by traditional depictions, e. g., in medical atlases non-photorealistic rendering (NPR) techniques [SS02] emerged to produce comprehensible renditions. NPR techniques range from those line drawings inspired from traditional artwork (such as silhouettes and hatching) to methods derived from DVR (such as tone shading). So far, mostly individual techniques were used—either traditional (e. g., DVR or surface rendering) or NPR methods. In this paper, we describe the integration of line drawings with surface and volume visualization to create more comprehensible renditions. In particular, we discuss the problems that arise when these are used together in an OPEN INVENTOR based scene graph architecture [Wer94].

We rely on segmented data where the relevant anatomic



(a) Too much transparency. (b) Too little transparency. (c) Badly differentiable.

Figure 1: *3D visualization with surface rendering and DVR: context structures (bones, kidneys, gallbladder, lung, and aorta) are not differentiable or hiding the focus. The focused liver includes a tumor, the portal vein, and the hepatic vein.*

structures are delineated. This makes it possible to generate renderings of particular objects. Applications of such hybrid visualizations lie in therapy planning and educational systems. In addition, hybrid visualizations are intended to explain a planned intervention to patients. For these applications, it is essential to employ patient-individual datasets that have a reduced resolution and a higher noise level than artificial datasets such as the VISIBLE HUMAN [SASW96].

In order to confirm the effectiveness of the created visualizations, we carried out a user study. In this study, we presented the renditions to different groups that represent the different possible potential users of the visualizations: medical professionals and medical laypersons.

The remainder of this paper is organized as follows. Section 2 describes related work. Afterwards, we discuss the combination of the three mentioned rendering methods in Section 3. Section 4 then presents the user study we carried out to compare the usefulness of the different visualizations. Finally, in Section 5 we conclude the paper.

2. Related Work

The main source of inspiration for our work comes from medical atlases (e. g., [Rog92]) where line primitives are often used to depict form and to distinguish objects from each other. This is motivated by research in perception. For example, WARE discusses that silhouette lines are essential for figure-ground segregation, for the fast recognition of objects, and for the identification of object structure [War04].

The development of non-photorealistic rendering styles [SS02] to generate images similar to those in medical atlases started approximately in 1990 [ST90]. It is interesting to note that NPR has very similar goals as scientific visualization, namely to convey information effectively and to emphasize features [NSW02]. Thus, it is not surprising that the visualization community adopts techniques from NPR. For the first time this has been done in a system for radiation treatment planning [LFP*90].

The term *non-photorealistic volume rendering*—or as RHEINGANS and EBERT suggest *volume illustration*—refers to visualization techniques that are similar to surface-based NPR but are applied to volume data [RE01]. In the context of medical volume data we prefer the term volume illustration because there is no photorealistic rendering option for a clinical CT dataset.

One class of volume illustration techniques operates on the original data without any segmentation information. The gradient direction is estimated based on differences between adjacent voxels. Silhouette lines are generated where the gradient magnitude is large and the cross product of camera direction and the gradient direction is close to 0. [CMH*01] presented expressive visualizations where the large gradient between air and skin is employed for silhouette generation. [LM02] introduced an efficient implementation of volume illustration that exploits commodity graphics hardware. However, for many regions in the human body contrasts in CT and MRI data is considerably lower than at the air-skin boundary and silhouettes cannot be identified easily.

[SE04] calculate silhouettes from volumetric data using implicit trivariate tensor product B-spline functions that approximate the data. A subdivision method is used to calculate the silhouettes that have a superior look compared to voxel-based silhouette extraction schemes. Convincing results were achieved recently by KINDLMANN et al. who employed curvature information to guide the placement of silhouette lines [KWTM03]. Transfer functions (TF) could be specified with the first and second main curvature as the two-dimensional domain for TF specification. They also generate contours and can control their thickness in image-space by considering curvature information.

Hatchings based on volume information allow to produce smoother hatching lines compared to purely surface-based methods [DCLK03]. [NSW02] introduce concepts for hatching volume data based on curvature information which is extracted near user-selected seed points. Hatching is modified by transparency values and lighting conditions. They argue not to combine DVR and line drawing in order to allow for flexible stylization.

For clinical applications it is more appropriate to rely on segmentation information and to deliberately assign rendering styles to anatomic structures or categories such as nerves or muscles. [HBH03] combined volume illustration techniques with surface rendering and DVR in a two-pass approach. This is achieved by applying local TFs to different objects using graphics hardware shader.

A recent approach presented by YUAN and CHEN [YC04] combines DVR with surface shading and also adds silhouettes and feature lines. However, their approach is based on image-space silhouette and feature line detection and is, therefore, limited to image-processing stylization. Our approach, in contrast, uses object-space stroke extraction that allows more freedom to parameterize the rendered lines.

[VKG04] also employed segmentation information in or-

der to specify the appearance of individual objects. In addition, they integrated illustration techniques such as cut-away views. The general concept of this work is to derive the importance of an object (e. g., from input by the user) and to project the most important objects onto the screen. All rendering styles are provided. However, they are integrated in a DVR which limits the flexibility of line stylization.

3. Combining Silhouettes, Surface Shading and DVR

In this section we describe a rendering process that allows to combine the three visualization techniques named above to create hybrid renditions. We integrate these techniques using an OPEN INVENTOR scene graph architecture which allows us to reuse nodes that affect visualization parameters at different positions in the same scene graph. This not only generates a more consistent visualization but also is more flexible in terms of stylizing individual objects.

3.1. Initial Considerations

The different rendering styles are not equally well suited to visualize objects. Thus, they will be used to depict different structures according to their relevance for the visualization (see Figure 7). With respect to relevance, in the following we will refer to three types of structures or objects:

Focus objects (FO): objects in the center of interest are emphasized in a particular way.

Near focus objects (NFO): important objects for the understanding of the functional interrelation or spatial location. Their visualization depends on the particular problem.

Context objects (CO): all other objects.

Line stylization—when carefully applied—may be used to control whether objects are conceived as FO or CO. One type of stylization that we include is color since this is widely used to emphasize objects or structures. In addition, we will depict hidden and visible lines in a different manner in order to convey the spatial relationships.

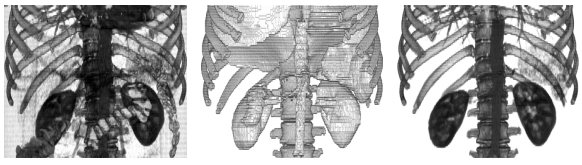
For the efficient extraction of lines and strokes from 3D meshes, a Winged Edge data structure is employed to store local connectivity information. This data structure has to be created as a pre-computation step for the individual objects. In order to render and stylize these objects, each object's mesh data is subsequently used to extract, e. g., the lines, and stylize them. Therefore, the use of stylization pipelines that successively modify the line style data is required. The stylization process is divided into several small steps represented by nodes in the scene graph. This concept also allows the reuse of certain line stylization pipelines because the respective nodes may be linked into the scene graph at several positions. This ensures a coherent appearance of the objects that use the same pipeline.

In order to use DVR, we integrate this rendering technique into one scene graph. This is achieved by specialized DVR node coupled with a TF node. However, since DVR always renders the whole volume into the z-buffer, it is not possible to add surface shading afterwards. Thus, the sequence in which the modules for the individual rendering techniques are added to the rendering pipeline will be important.

3.2. Hybrid Rendering

For surface shading the OPEN INVENTOR architecture relies on normal z-buffer rendering. Thus, no special order of nodes is required for a correct rendering of the scene. Due to the nature of the remaining two techniques, the resulting scene graph may get fairly complex. Therefore, we start by discussing the extension of the scene graph for surface shading to include DVR. Then, we will show how line rendering can be added and explain the required individual modifications. Our rendering process is based upon the scene graph architecture of OPEN INVENTOR. In addition, we use the OPENNPAR system that extends OPEN INVENTOR and adds line extraction and line stylization capabilities [HIR*03]. For easy scene graph manipulations as well as DVR, we employ the MEVISLAB system [HLP03].

Direct volume rendering. DVR may be combined with surface rendering by adding the respective nodes to the scene graph. However, the DVR node fills the z-buffer for the entire volume that is rendered regardless of the TF as explained above. Therefore, it has to be added to the scene graph after the surface shading has been completed, i. e., as the last node in the scene graph traversal. Otherwise, the surface objects would not be rendered because all would fail the z-buffer test. Hence, after the DVR, the z-buffer contains no more sensible depth information. Furthermore, DVR may previously be modified according to segmentation results as shown in Figure 2. The bit mask of segmented areas is used for the purpose of displaying or hiding user-selected objects.



(a) Original DVR. (b) Bitmask from segmentation. (c) Masked DVR.

Figure 2: *The original volume has been segmented in advance. Thus, it may be restricted to bones, gallbladder, aorta, kidneys, milt and lung. The gallbladder is not visible because of the chosen transfer function.*

Silhouette rendering. The object-space line rendering approach that we employ comprises the following steps: geometry data preparation, line extraction and stroke generation, hidden line removal, and stroke stylization (see Figure 3).

To facilitate the following stages, first a geometry data

structure is created that provides adjacency information. Then, after extracting the significant edges (silhouettes and feature lines), these edges are assembled into strokes and are processed in a stroke pipeline for stylization. As an essential step in this pipeline hidden lines have to be removed. A fast and simple method to solve this problem is illustrated in Figures 4(a) to (c). First, the objects are rendered into the z-buffer (while the frame buffer remains unchanged). In a second step, all extracted lines are scan-converted individually and stepwise classified as hidden or visible using the previously generated z-buffer data by locally comparing z-depths [IHS02]. Then, stylization may be applied such as changing the stroke's width, saturation, color, and texture. Further methods for line stylization such as cut away views or lighted regions may also be applied. In fact, the edge extraction and stroke generation is independent from the final stroke rendering because the data in the stroke pipeline is not altered by any of the other rendering steps. We will use this fact later on for combining all three rendering techniques.

If surface rendering is used in addition to the line graphics, the surface objects have to be rendered into the final image prior to the lines. Silhouettes being located exactly at discontinuities of the z-buffer of the surface objects. One side of each generated and stylized line would otherwise be overwritten by the surface object since they are typically more than one pixel wide. This is also the reason why there has to be an explicit hidden line removal (HLR) for the computed strokes before starting the final rendering process. In addition, the z-buffer generated for HLR would interfere with a correct rendering of the surface objects.

However, this approach is only applicable for opaque objects because transparent objects usually do not change the z-buffer. Thus, lines that lie behind a transparent object would not be removed. In order to prevent distant lines to be rendered on top of closer transparent model parts, the z-buffer rendering must be carried out for the transparent objects as well. Unfortunately, this conflicts with the regular shading technique for transparent objects.

Combination of rendering styles. According to the discussion above, the line extraction and visibility classification has to occur before rendering surface objects. Also, the DVR has to be performed after all surface objects have been drawn. However, rendering surfaces on top of the stylized lines would potentially overwrite parts of the rendered lines as explained above. Fortunately, we can make use of the line extraction, classification, and storage being independent from the process of rendering the line into the frame-buffer as suggested above. Therefore, we use the following procedure for generating the hybrid rendition:

1. generate the z-buffer for surface and line objects (including transparent objects),
2. extract lines from internal mesh representation,
3. determine line visibility according to the z-buffer,
4. clear the z-buffer,
5. render surface objects using z-buffering,
6. render stylized lines with writing z-buffer data but without doing the z-buffer test, and
7. render volume using z-buffering.

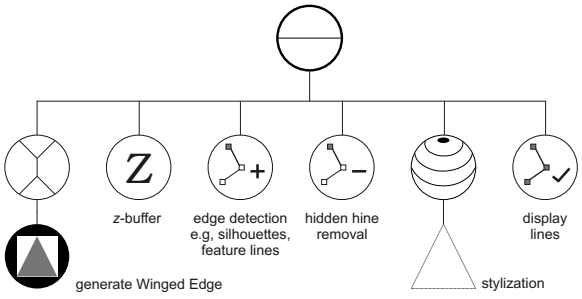
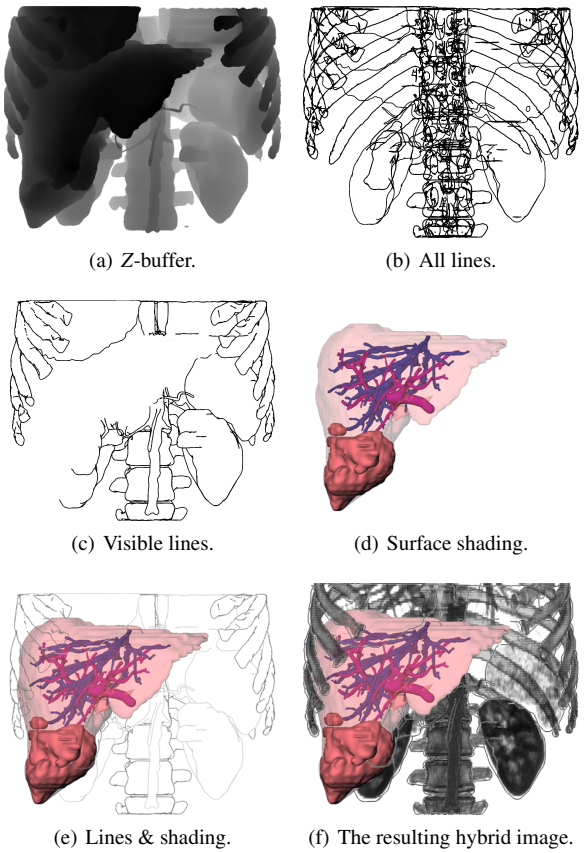


Figure 3: *Scene graph for silhouette rendering.*



(a) Z-buffer. (b) All lines. (c) Visible lines. (d) Surface shading. (e) Lines & shading. (f) The resulting hybrid image.

Figure 4: *Sequence for combining all rendering styles.*

The z-buffer of the surface objects and the line objects is rendered first (Figure 4(a)). The z-buffer is generated for all objects regardless whether they are transparent or opaque. Thereafter, the lines are generated (Figure 4(b)) and HLR is performed using the z-buffer information (Figure 4(c)). Because the line data is stored separately, it is not affected by

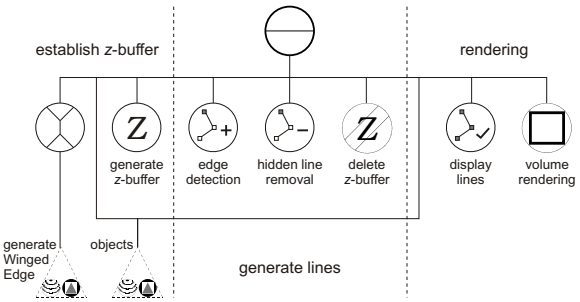


Figure 5: *Scene graph for combining all rendering styles.*

surface and volume rendering. Thus, it can be determined this early in the process. After the line extraction, the generated z-buffer is not needed anymore because it also contains transparent and line objects and becomes deleted.

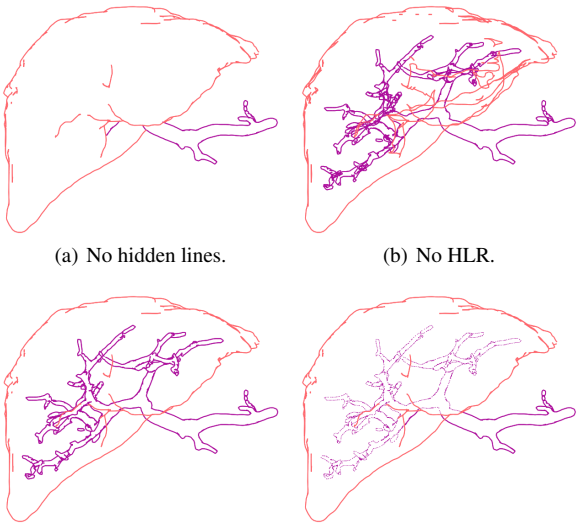
Now, the surface rendering is initiated (Figure 4(d)). Since we included transparent objects in the initial z-buffer rendering, there will be no lines that will mistakenly be drawn on top of them. Due to the separate storage of the stroke data, the lines can be displayed with correct depth information. For this purpose, the line rendering is performed without z-buffer test but with writing z-buffer data (Figure 4(e)). DVR is carried out as the last step and after the lines because now the line data is present in the z-buffer as well (Figure 4(f)). Figure 5 shows the scene graph for the entire process.

Removing self-occluding lines. In some cases it might be useful to show line drawings behind other line drawings. However, at the same time the hidden lines of both the front and back objects have to remain hidden so that the rendition does not get confusing. Unfortunately, the process discussed so far does not allow this. If both objects are rendered into the z-buffer simultaneously, it is not possible to distinguish between lines that are self-occluding and those that are hidden by a different object.

This problem can be solved by rendering the z-buffer for each object separately such that the self-occluding lines can be removed individually for each object (see Figure 6). The disadvantage of this procedure is that after each individual HLR it is necessary to clear the z-buffer. However, the produced effect (see Figure 6(d)) illustrates the spatial relationships and improves the understanding of the geometry.

4. Evaluation

In order to analyze whether line rendering is a meaningful extension to the existing visualization techniques, we carried out a user study. In addition to that, we have analyzed the answers to extract parameters for useful hybrid renditions combining DVR, surface shading, and line rendering. However, the goal for this task was not to find the 'best' visualization but several appropriate ones for the specific dataset since the quality of visualizations strongly depends on the domain.



(a) No hidden lines. (b) No HLR.

(c) Individual HLR. (d) Dashing for hidden lines.

Figure 6: *Removing self-occluding lines. In (a) the rendering of hidden lines is prevented due to a collective z-buffer. Individual HLR (c) solves this problem. Additional dashing and thinner hidden lines (d) produce an even stronger effect.*

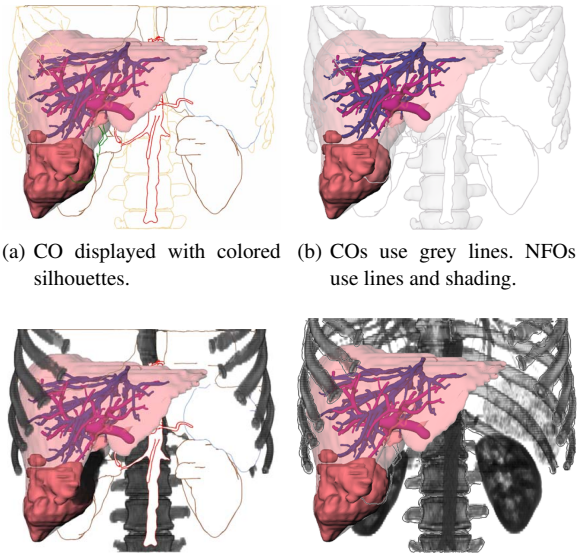
In cases where structures have to be displayed as COs, traditional illustrations typically use silhouettes and feature lines. We hypothesize that line renderings are preferred for the joint visualization of COs, NFOs and FOs. We also hypothesize, that a stylization with colors is most satisfying.

4.1. Evaluated Application Domains

The visualizations shown in this paper were generated for an educational system, the LIVERSURGERYTRAINER [BMOP04]. In this training system, users are asked, for example, to decide whether the patient could be safely operated and specify a resection line. For this purpose, comprehensible renderings of the intrahepatic anatomy (vascular structures, liver tumors) are necessary. In order to provide a realistic setting, the system employs clinical datasets. The relevant anatomic and pathologic structures are segmented to permit selective visualizations.

Context visualization. There are many possibilities for the visualization of COs. The shape of the structure to be visualized will finally prove the most appropriate variant (Figure 7). DVR usually looks unnatural and is difficult to interpret. On the other hand, DVR shows the original data and is able to show non-segmented structures.

Simplifying complex visualizations. In complex scenarios, only few viewing directions allow to appraise how all objects are situated to each other. By providing interaction means to view the scenario from all directions, viewers watch it also from other directions. By employing silhouettes we are able



(a) CO displayed with colored silhouettes.

(b) COs use grey lines. NFOs use lines and shading.

(c) NFOs use DVR while COs are shown using colored lines on a white background.

(d) COs and NFOs are shown using DVR. NFOs also have lines added.

Figure 7: *A number of possibilities to visualize COs, NFOs, and FOs differently. The FO is the liver in all visualizations. NFO are the bones and the gallbladder. Figure 7(c) and 7(d) were not part of the evaluation.*

to achieve a plainer visualization for suboptimal directions that is easier to understand.

Figure 8 illustrates this application. A liver is divided into several vascular territories. To eliminate a liver tumor, in most cases all involved territories have to be removed. For planning purposes all vascular territories, the tumor, and the portal vein have to be visualized. Typically, visualizations as in the right of Figure 8 are used. They can be simplified by rendering the affected or healthy vascular territories using colored silhouettes.

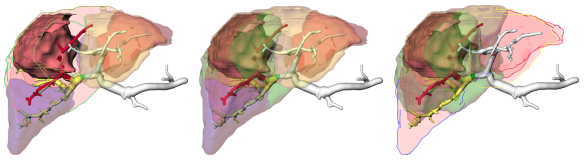
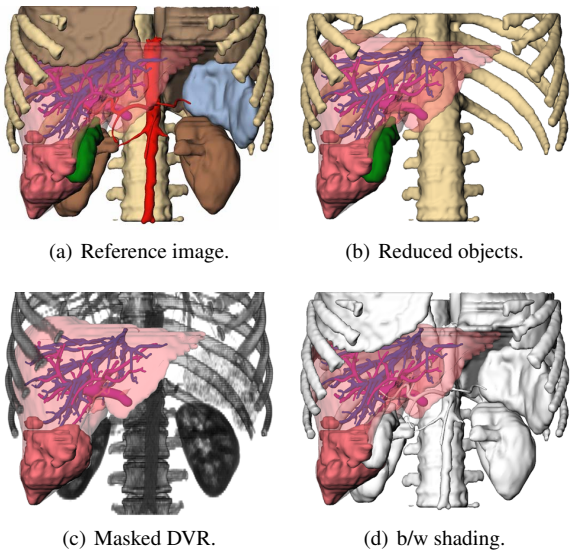


Figure 8: *Different coronal visualizations of the affected vascular territories. Left to right: affected vascular territories displayed via silhouettes, all segments transparent, and healthy territories via silhouettes.*

4.2. Study Subjects

The presented visualization process was developed for medical doctors who are familiar with medical atlases. Therefore



(a) Reference image.

(b) Reduced objects.

(c) Masked DVR.

(d) b/w shading.

Figure 9: *Selected variants without lines. Other visualizations shown are transparent versions of Figure (a) and (b).*

we asked them to compare and evaluate the quality of the visualizations. In addition, the visualizations were shown to medical laypersons. This second survey has practical relevance because patients usually receive a pre-surgery consultation. Comprehensible 3D visualizations may be employed in such a consultation.

4.3. Methodology

Based on a CT dataset, we generated a representative repertory of possible visualizations. In computer-assisted surgery, surface visualizations and DVR are common (see Figure 9). These visualizations were compared with hybrid renditions using silhouettes. Considering the pros and cons of each visualization for specific tasks such as using more or less context to show the FO, the subjects were asked for their preference. To obtain meaningful answers, we had to narrow down the specific application domain and chose liver surgery. To get as many as possible answers, the questionnaires were sent in printed form. The chosen viewing direction is the 'view of the surgeon', meaning the perspective of the surgeon on the patient during the surgery.

4.4. Questionnaire Assembly

Our questionnaire is guided by principles described in [Shn97]. Every page of the questionnaire uses the same pattern. On top, two visualizations of the same anatomic structures are shown. Below these images multiple choice questions are placed. First, one of the two presented pictures had to be chosen. This question asked for the personal preference without closer examination whether the image is particulary

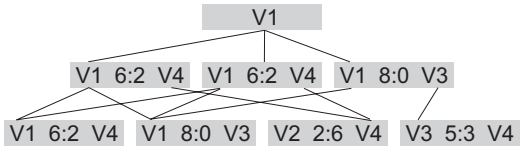


Figure 10: *Extraction of the decision tree for the medical doctors after critical examination. V1 refers to Figure 7(a), V2 to Figure 9(b), V3 to Figure 4(e), and V4 to Figure 9(a). From these Figure 7(a) was voted to be best.*

suited to solve certain problems. Subsequently, specific questions were asked to encourage the respondent to comment on the usefulness of the visualizations. In these questions a specific problem had to be assessed on a five value scale (for example ranging from 'very clearly arranged' to 'very complex'). Finally, the subjects were asked to specify the variant they would prefer for surgery education.

To classify the answers, we asked for some personal data: age, gender, education level, and personal skills with PCs as well as with 3D applications. The laypersons were introduced into the necessary medical knowledge by an additional preamble.

4.5. Analysis

For every pair of pictures on each page of the questionnaire the number of votes was counted. Due to the assembly, it could be determined which one of two images seems to be more suitable. The reference image (see Figure 9(a)) was compared with all other images. The remaining images were compared with only a few others to get a cross validation for the final results (shown in Figure 10). All in all 9 images for context visualization on 11 pages and 3 images for simplifying visualizations on one page are compared. Comparing all images with each other would have resulted in too many comparisons. By comparing the winners of each pair, we determined the most optical appealing and the most appropriate visualization for a given task since in medical training environments both aspects are important.

4.6. Interpretation

We received 33 filled out questionnaires in total. Eight sheets were returned by surgeons (2 female and 6 male) and 25 by medical laypersons (11 female and 14 male). The average age of the medical professionals is 42.8 years (ranging from 29 to 50) and they have medium skills with PCs and 3D applications. The average medical layperson is 25.3 years old (ranging from 23 to 27) and has very good skills with PCs and 3D applications.

Because only eight sheets were received by surgeons, it was difficult to extract significant results. In general, medical doctors settled upon less context information but noted that basic information about all COs has to be available all the time. In addition, the application of silhouettes seems to be appropriate for surgery planning. The visualization of the affected vascular territories using silhouettes was regarded as appropriate by six out of eight medical professionals. The two images that were favored by five of eight surgeons are shown in Figure 7(a) and 7(b). Our observations indicate a tendency, that surgeons prefer little context information. For this reason, the Figures 9(c) and 9(d) are also good candidates.

It was not possible to draw a distinct conclusion for the laypersons. This may be due to the novelty of the presented visualizations for the respondents. I. e., the reference image (Figure 9(a)) was compared to the DVR, the transparent, and the hybrid visualization. For the DVR and the transparent surface shading, no significant tendency was registered. Among all images which include silhouettes, 75% favored Figure 7(a). 83% favored Figure 7(a) compared to the transparent visualization. Almost all laypersons favored the silhouette image with additional surface shading.

Our results indicate that silhouettes are an appropriate extension of the existing visualization possibilities. In a direct comparison between the transparent surface and the hybrid visualization, the silhouettes are regarded as superior. The exclusive application of silhouettes without any further information about shape or color of the objects was regarded as unfavorable by all respondents. However, with additional information such as colored silhouettes or highly transparent surfaces, the silhouettes were rated to be very good.

5. Conclusion & Future Work

In this paper we described a scene graph based combination of silhouette, surface, and volume rendering. It shows that using a scene graph architecture facilitates a flexible and parameterizable integration of these existing rendering components. Stroke extraction and stroke rendering are decoupled from each other to allow for the display of correctly determined visible lines with stylization applied. Existing techniques such as using one or several semi-transparent layers of shaded objects may be used as well. One limitation is that we cannot render semi-transparent shading with the VR shining through. This would require to render the volume first which is not possible with the proposed pipeline. The combination of silhouette rendering with the traditional rendering styles for medical visualization has great potential. The integration of silhouettes allows to convey complex spatial relations more clearly which is essential for therapy planning and medical education. We collected feedback from medical professionals and laypersons indicating that hybrid renderings including silhouettes are appropriate to convey spatial information based on clinical data.

A great deal of work still needs to be done. Illustration techniques such as hatching and stippling should be integrated in a similar object-based manner. The interaction to

adjust hybrid medical visualizations should be redesigned to reduce the interaction effort. In [KTH*05], applications of hybrid renderings for a special area of surgery planning are described.

Acknowledgments

Special thanks go to Angela Brennecke and Sebastian Mirschel. The liver data sets were provided by Prof. Dr. Karl J. Oldhafer of the Allgemeines Krankenhaus Celle, Germany. Prof. Oldhafer substantially supported the user study. Thanks also to Wolf Spindler for providing the volume renderer. The high-quality vessel visualizations in this paper are created by [OP05]. This work was carried out in the framework of a project supported by the Deutsche Forschungsgemeinschaft (DFG) (Priority Programme 1124, PR 660/3-1).

References

[BMOP04] BADE R., MIRSCHEL S., OLDHAFER K. J., PREIM B.: Ein fallbasiertes Lernsystem für die Behandlung von Lebertumoren. In *Bildverarbeitung für die Medizin* (2004), Springer, pp. 438–442.

[CMH*01] CSÉBFALVI B., MROZ L., HAUSER H., KÖNIG A., GRÖLLER E.: Fast Visualization of Object Contours by Non-Photorealistic Volume Rendering. *Computer Graphics Forum 21*, 3 (Sept. 2001), 452–460.

[DCLK03] DONG F., CLAPWORTHY G. J., LIN H., KROKOS M. A.: Nonphotorealistic Rendering of Medical Volume Data. *IEEE Computer Graphics & Applications 23*, 4 (July/Aug. 2003), 44–52.

[HBH03] HADWIGER M., BERGER C., HAUSER H.: High-Quality Two-Level Volume Rendering of Segmented Data Sets on Consumer Graphics Hardware. In *Proc. of IEEE Visualization* (2003), pp. 301–308.

[HIR*03] HALPER N., ISENBERG T., RITTER F., FREUDENBERG B., MERUVIA O., SCHLECHTWEG S., STROTHOTTE T.: OpenNPAR: A System for Developing, Programming, and Designing Non-Photorealistic Animation and Rendering. In *Proc. of Pacific Graphics* (2003), IEEE, pp. 424–428.

[HLP03] HAHN H. K., LINK F., PEITGEN H.-O.: Concepts for Rapid Application Prototyping in Medical Image Analysis and Visualization. In *Simulation und Visualisierung* (2003), SCS, pp. 283–298.

[IHS02] ISENBERG T., HALPER N., STROTHOTTE T.: Stylizing Silhouettes at Interactive Rates: From Silhouette Edges to Silhouette Strokes. *Computer Graphics Forum 21*, 3 (Sept. 2002), 249–258.

[KTH*05] KRÜGER A., TIETJEN C., HINTZE J., PREIM B., HERTEL I., STRAUSS G.: Interactive Visualization for Neck-Dissection Planning. In *Proc. of EuroVis* (2005), Eurographics Association.

[KWTM03] KINDLMANN G. L., WHITAKER R., TASDIZEN T., MÖLLER T.: Curvature-Based Transfer Functions for Direct Volume Rendering: Methods and Applications. In *Proc. of IEEE Visualization* (2003), pp. 513–520.

[LFP*90] LEVOY M., FUCHS H., PIZER S. M., ROSENMAN J., CHANEY E. L., SHEROUSE G. W., INTERRANTE V., KIEL J.: Volume Rendering in Radiation Treatment Planning. In *Visualization in Biomedical Computing* (1990), pp. 4–10.

[LM02] LUM E. B., MA K.-L.: Hardware-Accelerated Parallel Non-Photorealistic Volume Rendering. In *Proc. NPAR* (2002), ACM Press, pp. 67–74.

[NSW02] NAGY Z., SCHNEIDER J., WESTERMANN R.: Interactive Volume Illustration. In *Proc. of Vision, Modelling and Visualization* (2002), pp. 497–504.

[OP05] OELTZE S., PREIM B.: Visualization of Vascular Structures: Method, Validation and Evaluation. *IEEE Transactions on Medical Imaging 24* (2005). To appear.

[RE01] RHEINGANS P., EBERT D.: Volume Illustration: Nonphotorealistic Rendering of Volume Models. *IEEE Transactions on Visualization and Computer Graphics 7*, 3 (July–Sept. 2001), 253–264.

[Rog92] ROGERS A. W.: *Textbook of Anatomy*. Churchill Livingstone, 1992.

[SASW96] SPITZER S., ACKERMAN M., SCHERZINGER A., WHITLOCK D.: The Visible Human Male: A Technical Report. *Journal of the American Medical Informatics Association 3*, 2 (Mar./Apr. 1996), 118–130.

[SE04] SCHEIN S., ELBER G.: Adaptive Extraction and Visualization of Silhouette Curves from Volumetric Datasets. *The Visual Computer 20*, 4 (June 2004), 243–252.

[Shn97] SHNEIDERMAN B.: *Designing the User Interface*, 3rd ed. Pearson Addison Wesley, 1997.

[SS02] STROTHOTTE T., SCHLECHTWEG S.: *Non-Photorealistic Computer Graphics. Modelling, Animation and Rendering*. Morgan Kaufmann, 2002.

[ST90] SAITO T., TAKAHASHI T.: Comprehensible Rendering of 3-D Shapes. In *Proc. of SIGGRAPH* (1990), ACM Press, pp. 197–206.

[VKG04] VIOLA I., KANITSAR A., GRÖLLER M. E.: Importance-Driven Volume Rendering. In *Proc. of IEEE Visualization* (Los Alamitos, 2004), IEEE, pp. 139–145.

[War04] WARE C.: *Information Visualization*, 2nd ed. Morgan Kaufmann, 2004.

[Wer94] WERNECKE J.: *The Inventor Mentor: Programming Object-Oriented 3D Graphics with Open Inventor, Rel. 2*. Addison-Wesley, 1994.

[YC04] YUAN X., CHEN B.: Illustrating Surfaces in Volume. In *Proc. of VisSym* (2004), Eurographics Association, pp. 9–16.

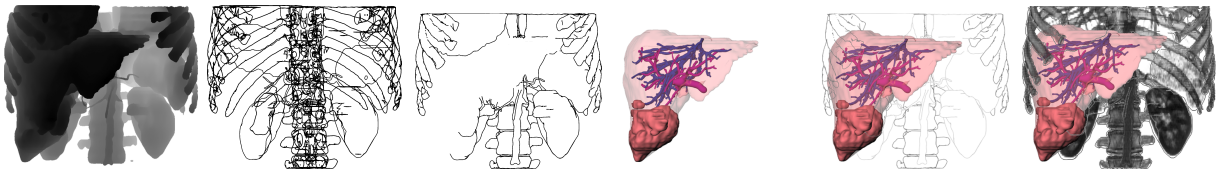


Figure 4: *Sequence for combining all rendering styles.*

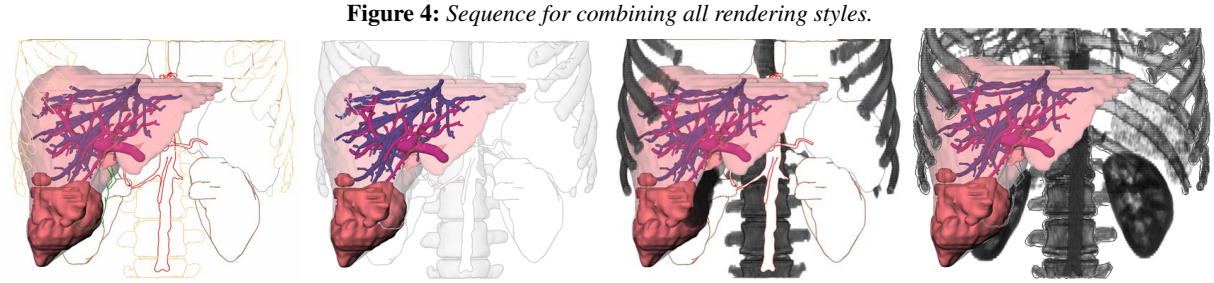


Figure 7: *A number of possibilities to visualize COs, NFOs, and FOs differently.*

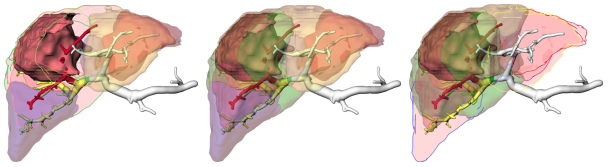


Figure 8: *Different coronal visualizations of the liver.*

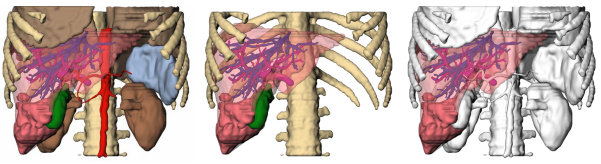


Figure 9: *Selected variants without lines.*