

Virtual Resection with a Deformable Cutting Plane

Olaf Konrad-Verse¹, Bernhard Preim², Arne Littmann¹

¹Center for Medical Diagnostic Systems and Visualization, Universitaetsallee 29,
28359 Bremen, e-Mail: okonradlittmann@mevis.de

²Institute for Simulation and Graphics, Department for Computer Science
Otto-von-Guericke-University of Magdeburg, Universitaetsplatz 2, 39106 Magdeburg
e-Mail: preim@isg.cs.uni-magdeburg.de

Abstract. We describe methods for the specification and modification of virtual resections in medical volume data. These techniques are focused on applications in therapy planning, but are also relevant for educational purposes. The basic idea of our approach is to convert resection lines specified by the user on an organ's surface into an initial mesh by means of a principal component analysis. This mesh can be flexibly deformed with an adjustable sphere of influence. We present applications in liver surgery planning.

Keyword: 3d interaction, clipping, computer-assisted surgery

1 Introduction

The ability to specify virtual resections (those parts of the body which should be removed later in surgery) is a core function of many intervention planning systems. In difficult cases, the exploration of resection strategies directly supports the question whether or not the resection is feasible at all and the specification of the resection strategy. The specification of a virtual resection can be used for a quantitative analysis with respect to the volume of the intended resection or the percentage of an organ which would be removed.

Technically, a user must be able to specify an arbitrarily-shaped 3d clip geometry as resection volume and the clip geometry should be rendered fast and accurate. Modern APIs, such as OpenGL, and graphics hardware, however, only support clipping geometries composed of clipping planes. Recent work by Weiskopf et al. tackled the visualization and rendering problems associated with complex clip geometries [1]. They integrated the tests to consider complex clip geometries in texture-based volume rendering and also described methods of realistic shading of the resulting clip surfaces. The focus of this paper is the description of 3d interaction techniques to specify and to modify the clip geometry with medical doctors as intended users. The resection planning module has to fulfil several requirements:

- (1) The user must be able to specify a virtual resection intuitively and precisely.
- (2) Modification facilities are required to change virtual resections either to correct them or to explore other variants.
- (3) Virtual resections should be visualized immediately in high quality.

The first requirement means that medical doctors should be able to decide in detail which anatomic structures will be removed. For example, the question whether or not a certain vascular structure will be cut is often essential, because it may have a strong effect on the whole procedure, in particular if a blood vessel would have to be reconstructed. The modification of a virtual resection is crucial in order to support a flexible exploration of the resection facilities and also to correct undesired effects which occur rather often. It is essential that virtual resections might be modified on different scales. The third requirement refers to the update and rendering mechanism which is used to show the effect of a virtual resection. This part has to be considered since medical volume data are large and the update should happen with interactive frame rates.

2 Prior and Related Work

There are several possibilities how a virtual resection might be specified. Some methods are restricted in the shape of resections which can be produced. For example, the extrusion of shapes, such as prisms, along a path cannot serve as general approach to define arbitrarily shaped resection volumes. In medical imaging such techniques are primarily provided to remove irrelevant structures and to explore data. In the following we focus on real-time interaction techniques which can produce any desired shape of the clip geometry. After discussing related work by other groups in Sect. 2.1, we briefly describe two approaches to virtual resection which we have investigated in the past (Sect. 2.2. and 2.3).

2.1 Volume Sculpting, Volume Cutting and Surgery Simulation

There has been considerable effort to allow volumetric modelling in a similar way like geometric modelling with surface representations. This work is often summarized under the term *volume sculpting*. One approach is to modify volumetric data by adding or removing certain graphical primitives. In traditional geometric modelling this is referred to as constructive solid geometry (Wang and Kaufman [2]). Another approach is to allow the user to cut into the data (*volume cutting*) and to continuously update the visualization. Volume cutting is often accomplished in connection with medical volume data and the interaction is inspired by real surgical procedures. Pflesser et al. [3] present the interaction with a virtual scalpel and with drilling devices in order to simulate surgery of the petrous bone (surgery in the ear region). The strongest aspect of this work is that the cut surfaces are accurately depicted, so that even zoomed in versions of sharp edges are displayed in high quality by computing the surface at subvoxel resolution. This is achieved by adaptive subsampling in the regions where the cutting device intersects the volume data. With this approach the cutting device (represented as surface description) is converted into voxels at a resolution above the medical volume data. For surgery simulation, volume cutting is often combined with haptic feedback in order to simulate not only the look of the data but also the feel (Avila and Sobierajski [4]). For the education of surgeons a realistic tactile feedback is essential.

There are some differences and some similarities with the requirements our works targets at. First, our virtual resection method is intended for experienced surgeons

which actually plan a surgical procedure. Therefore, we do not rely on specialized hardware, such as those for tactile feedback. In general, our focus is not the realistic simulation of a procedure but on decision support based on the interaction with the data of a particular patient. A realistic simulation of an operation with a virtual scalpel for example takes too long for the use in a clinical environment. The similarities between the above-mentioned work and our problem concern the representation and the visualization of the data. We need an additional volume dataset to represent what has been virtually resected. This is necessary to reverse virtual resections and to operate with medical volume data without changing it. Also, we must be aware of the update rate. Virtual resections only affect a small amount of the overall data (recall [3]). In order to achieve a fast update of the visualization, the rendering should be restricted to the involved region.

2.2 Specification of Virtual Resections by Erasing

In previous work, we developed a method where scalable 3d shapes are used as *erasers* which remove the tissue which is touched [5]. With two-handed input, one hand to rotate the visualization and one to specify the resection, a multi-level undo-mechanism and special 3d image processing operations to post-process the resection volume an elaborate system was created. The 2d visualization in this approach was only used to present the result of the virtual resection. However, the evaluation clearly showed that it was too difficult to specify resections precisely enough. For example, surgeons often plan their resections in relation to bifurcations of a vascular system. This means that the virtual resection facility should enable the surgeon to specify that a certain segment of a vascular system will be resected. The specification by erasing turned out to be too difficult to control for such a fine-grained specification.

2.3 Specification of Virtual Resections by Drawing in Slices

The second approach to the specification of virtual resections is inspired by the communication between surgeons and radiologists discussing a resection: some slices of the CT data are copied and the resection is marked by drawing in the slices. Clearly, by drawing in slices a resection can be specified as precisely as desired. However, this process is time-consuming if the entire resection volume should be specified because often some 50-100 slices are involved. The process can be strongly enhanced with interpolation methods. Instead of drawing in all slices, the user might skip many slices in which the contour is computed by shape-based interpolation. For liver surgery planning, it turned out that the manual specification in 3 to 5 slices is often sufficient. While the interaction with this approach is restricted to 2d, the 3d visualization is very useful to evaluate the result and to discuss the shape of resections (complex shapes are difficult to realize surgically and would lead to long resections with more loss of blood). This second approach is part of our system INTERVENTIONPLANNER which is regularly used by medical doctors [6].

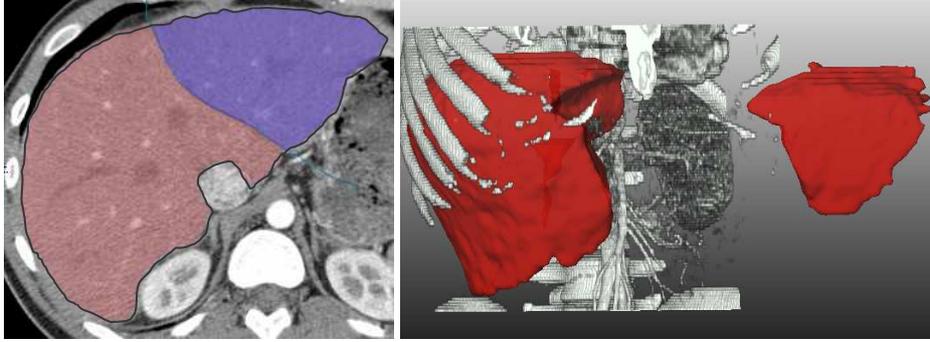


Fig. 1: Virtual liver resection by drawing in the slices. Using the liver segmentation mask, only one line per slice has to be drawn cross over the organ. In the 2d view (left image) the contour has been specified in selected slices. After the interpolation the result is displayed in the 3d view. It is translated to support the evaluation of the shape of virtual resections.

3 Virtual Resection with a Deformable Cutting Plane

The approach we present here has some similarities with the virtual resection based on erasing in the data (it includes 3d interaction) and some similarities with the virtual resections by drawing in the slices. The user should draw lines on the surface of an organ to initialize the virtual resection. Out of these lines, a mesh is generated that represents the initial cutting plane. The plane is deformed locally to fit the lines drawn by the user. This cutting plane can be interactively modified to refine the virtual resection.

3.1 Generation of the Initial Cutting Plane

The following algorithm describes how the lines drawn by the user are converted into an initial cutting plane.

1. *Determine the oriented bounding box of the lines drawn by the user.* This is accomplished by means of a principal component analysis of the point-set P forming the lines. Let P be the pointset and v the center of gravity (Eq. 1):

$$P = \{p_0, p_1, \dots, p_n\} \subseteq \mathbb{R}^3$$

$$v = \frac{1}{n+1} \sum_{i=0}^n p_i \quad (\text{Eq. 1})$$

The center of gravity is subtracted from all points p_i resulting in a new pointset P^{norm} . Based on this point set the covariance matrix A is determined (Eq. 2).

$$A = COV(P^{norm}, P^{norm}) \quad (\text{Eq. 2})$$

$$A \cdot c_i = \lambda_i \cdot c_i \quad (\text{Eq. 3})$$

A is a symmetric 3×3 matrix. Due to this property A has 3 Eigenvectors c_i and 3 Eigenvalues $0 \leq \lambda_0 \leq \lambda_1 \leq \lambda_2$ (cf. Eq. 3). The Eigenvectors define a local right-angled coordinate system of the pointset. Based on the normalized Eigenvectors a transformation matrix M_{rot} can be specified which describes the mapping from the pointset in the local coordinate system (Eq. 4). With this rotation, the original point set P can be transformed to local coordinates with respect to the cutting plane (Eq. 5).

$$M_{rot} = (|c_o \rangle, |c_1 \rangle, |c_2 \rangle) \quad (\text{Eq. 4})$$

$$y = M_{rot} \cdot p - v \quad (\text{Eq. 5})$$

2. *Determine the orientation and extent of the cutting plane.* The vectors (c_1, c_2) corresponding to the two largest Eigenvectors (λ_1, λ_2) resulting from the PCA define the cutting plane's orientation. The plane is determined by the following equation, where x represents any vector which satisfies Eq. 6.

$E : (v + x) \otimes c_0 = 0$ (Eq. 6), where \otimes denotes the dot product (see Fig. 2).

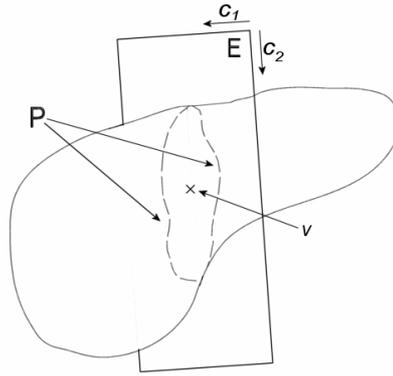


Fig. 2: Definition of the plane E based on the lines drawn by the user.

3. The center of the cutting plane (cp) is chosen such that it corresponds to v , the centre of gravity of P . This plane is then divided into a regular grid with quadrilateral cells. The resolution of the mesh is adjustable by the user. The default value is chosen such that the requirements of surgeons to specify typical resection shapes can be satisfied. A resolution with a cell size corresponding to $5 \text{ mm} \times 5 \text{ mm}$ in the underlying data proved satisfactory for resection planning in oncological liver surgery. The default size of the mesh is determined by the two largest Eigenvectors; the mesh is twice as long. Formally we can specify the cells of the mesh as follows.

$$y_{m,n} = m \cdot k \cdot c_1 + n \cdot k \cdot c_2 \quad (\text{Eq. 7})$$

4. Project Y into E.

$$y_i^P = (1 \quad 1 \quad 0) \cdot y_i \quad (\text{Eq. 8})$$

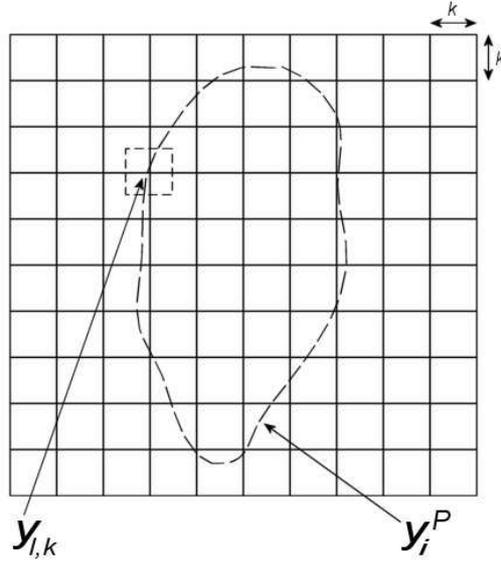


Fig. 3: Grid and spacing on the plane after projection.

5. *Calculate displacements.* The mesh is displaced in the c_0 direction to fit the original point set. This is accomplished by scanning all grid cells $y_{m,n}$ and testing whether any point of the transformed points y_i is projected in this cell. Fig. 3 shows that only a small subset of the cells are affected by the y_i . If this is true, the displacement $d_{m,n}$ is chosen such that it is the maximum distance of y_i to the plane E (see Fig. 4).

$$d_{m,n} = \|z_i\|_{\max}, \text{ (Eq. 9) where the } z_i \text{ represent the } z\text{-component of the } y_i$$

$$z_{m,n}^0 = y_{m,n} + d_{m,n} \cdot c_0 \text{ (Eq. 10)}$$

As an alternative, $d_{m,n}$ could be set to the average distance of the relevant points. The maximum, however, seems better suited because strong bulges specified by the user are better represented.

6. *Smoothing.* After these distances have been calculated for the whole mesh they are iteratively smoothed (laplace smoothing: the distance for one grid point is replaced by the weighted average of the distance of neighbouring points). As default, 100 iterations are carried out and the 3×3 neighbourhood is considered ($s=1$ in Eq. 11). However, the smoothing is only applied to those grid points which have not been displaced in step 5. This prevents that the points specified by the user are translated.

$$z_{m,n}^r = \alpha \cdot \frac{1}{(n+1)^2} \cdot \sum_{i=-s}^s \sum_{j=-s}^s z_{m+i,n+j}^{r-1} + (1-\alpha) \cdot z_{m,n}^r \text{ (Eq. 11)}$$

The mesh is rendered as a polygonal surface which appears smooth due to the Gouraud shading. The user can switch between two 3d visualization modes, one showing the

complete scene, while the other one is focussed on the structures of interest, namely the tumor, the vessels and the deformable cutting plane.

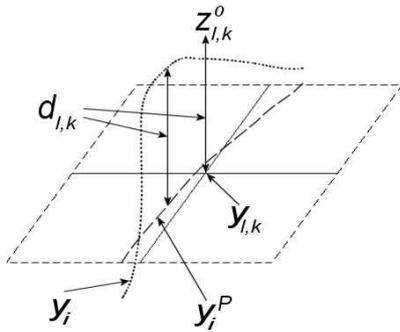


Fig. 4: The points are displaced orthogonal to E to reflect the position of the lines drawn by the user.

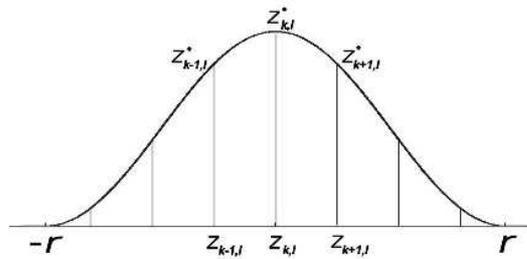


Fig. 5: The influence of the displacement initiated by the user has an influence with an adjustable radius r . Within this range the cosine function controls the amount of displacement.

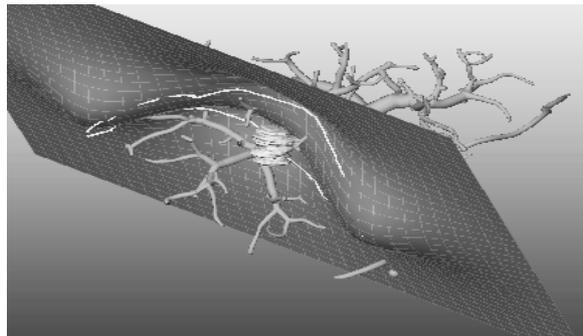


Fig. 6: Based on the two lines drawn on the object surface an initial resection has been specified which might be refined by the user.

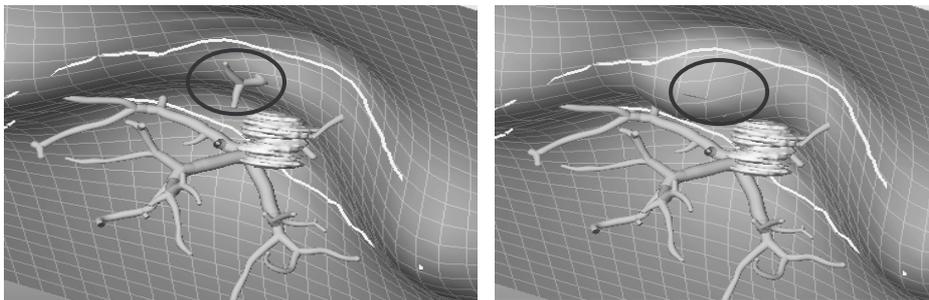


Fig. 7: The branch above the tumor (left image) is excluded from resection by locally deforming the virtual cutting plane (right image), see the encircled regions.

3.2 Modification of Virtual Resections

The resection can be refined by modifying the grid (translating grid points). The user can define the sphere of influence as well as the amplitude of the deformation. Both functions are controlled with a direct-manipulative style (mouse movements at the point which is modified). Usually, the deformation direction is orthogonal to the plane. This default behaviour can also be corrected. Within the sphere of influence, the cosine is employed to determine the extent of the displacement (the extent of the displacement decreases from a maximum at the selected point to zero at the border of the sphere of influence, see Fig. 5). Besides the local modification of the grid, there is also a facility to translate the whole mesh.

Application in surgery planning. As only the tumor and the vascular systems are essential for the decision where to cut, the organ surface can be hidden at this stage. Fig. 6 presents an example of an initial plane defined for liver surgery planning. In Fig. 7 this plane is modified based on information concerning vascular structures.

Of course, the surgeon can visually check the resection in 2d as well. Despite the intuitive clearness of 3d visualization, this is indispensable since sometimes not all important vascular branches can be segmented due to low grey value contrast. The planning process is completed by a volume calculation of both the resection and of the remaining part of the liver, which is an important criterion for estimating the operation risk.

General remarks. Besides these specific aspects there are some general facilities to modify virtual resections. As a general concept, an undo-facility is essential because inadvertent changes occur rather often. The second general strategy is to make use of both the 2d and the 3d visualization for the modification of a virtual resection. To modify the contour of the resection volume in a 2d slice view, requires converting the sequence of pixels into a parametric curve which can be manipulated by translating control points. As it is tedious to modify the contour in each slice, an interpolation algorithm is employed. The contours between the slices modified by the user are deleted and interpolated as discussed in Sect. 2.3.

3.3 Discussion

The virtual resection is typically carried out after several image analysis steps (segmentation of pathologic structures, vascular systems and organs). The segmentation for the respective organ is required for the virtual resection with the deformable cutting plane. The initial mesh for the virtual resection with the deformable cutting plane is based on a PCA of the points drawn by the user. An alternative would be to use linear regression to fit a plane to the points (for example by minimizing the least square error). The advantage of the PCA method is that not only the plane is defined but also the extent of the face can be derived with respect to the Eigenvalues. The PCA also results in a local coordinate system for the plane which is employed to determine a grid on the face with cells parallel to the border of the face.

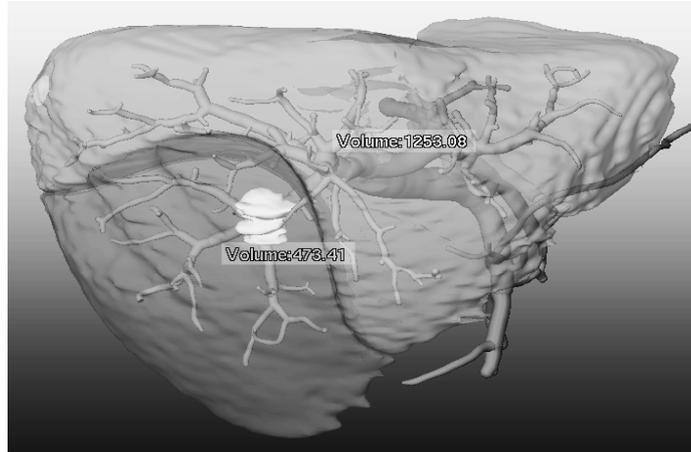


Fig. 8: The result of a virtual resection by means of a deformable mesh, including information about the volume of the resection and of the remaining liver tissue.

3.4 Efficient Visualization of Virtual Resections

The requirement to effectively update the display is a typical visualization problem. The resection volume has to explicitly represented in a similar way as segmentation results are stored in tagged volumes. A very fast approach has been described by Westermann et al. [8]. It is based on the graphics hardware, in particular it relies on the stencil buffer to decide which voxels should be included in the visualization. We choose a different method solely realized in software. We employ a page-based visualization (a page is a subvolume). In general, only those pages which are affected by the local modification of the resection plane are redrawn. For this purpose, the intersection between all modified triangles of the deformable cutting plane with page boundaries to decide which pages have to be redrawn. The resection plane itself is also voxelized and internally represented in the same way as any other segmented object.

3.5 Visualization Parameters for Resection Volumes

Once a virtual resection is specified it can be displayed in different ways. The realistic approach – to remove the resection area entirely – is only one solution. The resection volume can be regarded as a new visualization object to which visualization parameters can be assigned flexibly. In Fig. 8, for example, it is rendered semitransparently. Other objects, such as vascular structures, in the resection area are not affected. More general, it might be specified whether objects are clipped against resection volumes or not (this flexibility was first used for anatomy education and called *selective clipping*). For tumors and vascular structures, it is often useful to display them also in resection volumes. Again such behaviour can be specified as a default value for a category of anatomic objects.

4 Discussion

As an initial attempt to understand how surgeons use the method described here - an informal evaluation - has been carried out. Some comments of surgeons reveal desirable refinements. For example, the distance from the resection plane to a tumor should be continuously displayed to indicate the safety margin associated with the virtual resection. Another function which is desirable is to render the deformable cutting plane semitransparently. With this feature it might be easier to achieve the desired modification of the plane.

Usually, the initial viewing direction when drawing on the organ's surface is not appropriate to draw the line on the entire organ. Currently this process can be interrupted (by pressing "ESC") to rotate the virtual camera. This process (drawing, rotating, drawing) can be refined with two-handed input, where one hand (and input device) is used to control the virtual camera, and the other for the specification of the virtual resection. People are very effective in coordinated movements with both hands and there are successful applications of this principle in surgery planning (see Hinckley [9, 10] for an application of two-handed input in neurosurgery planning).

Combination of Resection Proposals and Virtual Resection. A completely different approach to resection specification is to propose the surgeon which part of an organ should be resected taking into account the position and size of a pathology and the position and structure of important adjacent structures. Tumor surgery for example attempts to achieve a 10 mm safety margin around a tumor which is completely removed. On the other hand, the remaining tissue should be supplied by blood vessels. We introduced such resection proposals for liver surgery, where we computed the vascular territories around a tumor with a certain safety margin [7]. The territories which have been calculated and presented as resection proposals turned out to be a good orientation for surgeons. However, the shape of the resection volumes was often too complex to realize the proposed resections precisely.

While the purely interactive specification of a virtual resection, for example with the deformable cutting plane, does not consider safety margins and blood supply, a combination of both is desirable. As a first step, the resection proposal might be presented as additional information when the deformable cutting plane is specified. A closer integration of both methods would convert the resection proposal in an initial cutting plane which is subsequently modified.

Another way to combine the "intelligent" resection proposals and the interactive resection specification is to initiate an analysis of vascular structures when the cutting plane is modified. As a result, all branches of vascular structures which are either in the resection volume or are dependent from those branches are highlighted to indicate the consequences of the intended resection. This analysis can be accomplished with the precisely the same methods which are employed for the generation of resection proposals.

5 Concluding Remarks

Virtual resection is an essential feature for surgery planning in particular for organs, such as kidney, liver and pancreas. We described a method which is based on a deformable cutting plane. The user can precisely control the orientation of the plane and the displacements orthogonal to the plane by drawing lines on an organ's surface. Fine-tuning facilities on different scales are provided to support precise interaction. While the virtual resections based on drawing in the slices are clinically used, the use of the deformable cutting plane is in an experimental stage. Therefore, a careful evaluation is needed to compare these methods. Initial feedback of surgeons shows that the deformable cutting plane is a promising approach for specifying virtual resections. This is also due to the fact that real surgical procedures are somehow simulated with this approach. If the surrounding tissue is removed surgeons mark the resection line on an organ's surface which is very similar to the specification of the initial resection face.

While this work is inspired by and targeted at clinical applications it might be relevant to other areas of interactive 3d visualization as well. As Weiskopf et al. point out, the exploration of 3d vector fields also benefits from volume clipping with complex clip geometries (recall [1]). We agree with Weiskopf et al. that volume clipping has a similar importance for the exploration and analysis of volume data than transfer function design which has gained considerable attention in the last years. While transfer function design is based on data (intensity values, gradient magnitude, ...), volume clipping is more intuitive because the interaction is related to the geometry.

Acknowledgements. We want to thank our collaborators at MeVis: Dr. H. Bourquain, A. Hennemuth, M. Hindennach, A. Schenk, and Prof. H.-O. Peitgen. The paper is also based on many discussions with our clinical partners: we want to thank Prof. H. Feussner (University Munich), Prof. K. Oldhafer (General Hospital Celle). In particular, Prof. Oldhafer provided valuable feedback on the approach described here.

References

- [1] D Weiskopf, K Engel, T Ertl (2003). „Interactive Clipping Techniques for Texture-Based Volume Visualization and Volume Shading”, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 9(3), pp. 298-312
- [2] S W Wang, A E Kaufman (1995). “Volume sculpting”, *Symposium on Interactive 3D Graphics*, pp. 151–156, ACM SIGGRAPH
- [3] B Pflesser, A Petersik, U Tiede, K H Höhne, R Leuwer (2002). “Volume Cutting for Virtual Petrous Bone Surgery”, *Computer-Aided Surgery*, Vol. 7(2), pp. 74-83
- [4] R S Avila, L M Sobierajski (1996). “A Haptic Interaction Method for Volume Visualization”, *IEEE Visualization '96*, pp. 197-204
- [5] B Preim, W Spindler, KJ Oldhafer, H-O Peitgen (2001). “3D-Interaction Techniques for Planning Oncologic Soft Tissue Operations“, *Proc. of Graphics Interface* (June 2001), Morgan-Kaufman, pp. 183-190
- [6] B Preim, M Hindennach, W Spindler, A Schenk, A Littmann, HO Peitgen (2003). “Visualisierungs- und Interaktionstechniken für die Planung lokaler Therapien“, *Proc. of Simulation and Visualization 2003*, SCS, pp. 237-248
- [7] B Preim, H Bourquain, D Selle, HO Peitgen, KJ Oldhafer (2002). “Resection Proposals for Oncologic Liver Surgery based on Vascular Territories”, *Computer Assisted Radiology and Surgery (CARS)*, pp. 353-358

- [8] R Westermann, T Ertl (1998). "Efficiently Using Graphics Hardware in Volume Rendering", *Proc. of SIGGRAPH*, pp. 169-179
- [9] K Hinckley, R Pausch, J C Goble, N F Kassell (1994). "Passive Real-World Interface Props for Neurosurgical Visualization", *Proc. of ACM Conference on Human Factors in Computing Systems*, pp. 452-458
- [10] J C Goble, K Hinckley, R Pausch, J W Snell, N F. Kassell (1995). "Two-Handed Spatial Interface Tools for Neurosurgical Planning", *Computer*, Vol. 28(7), pp. 20-26