LiveSync++: Enhancements of an Interaction Metaphor

Peter Kohlmann*
Vienna University of Technology

Stefan Bruckner†
Vienna University of Technology

Armin Kanitsar‡
AGFA HealthCare

M. Eduard Gröller§
Vienna University of Technology

ABSTRACT

The LiveSync interaction metaphor allows an efficient and non-intrusive integration of 2D and 3D visualizations in medical workstations. This is achieved by synchronizing the 2D slice view with the volumetric view. The synchronization is initiated by a simple picking on a structure of interest in the slice view. In this paper we present substantial enhancements of the existing concept to improve its usability. First, an efficient parametrization for the derived parameters is presented, which allows hierarchical refinement of the search space for good views. Second, the extraction of the feature of interest is performed in a way, which is adapting to the volumetric extent of the feature. The properties of the extracted features are utilized to adjust a predefined transfer function in a feature-enhancing manner. Third, a new interaction mode is presented, which allows the integration of more knowledge about the user-intended visualization, without increasing the interaction effort. Finally, a new clipping technique is integrated, which guarantees an unoccluded view on the structure of interest while keeping important contextual information.

Keywords: Smart Interaction, Linked Views, Medical Visualization, Viewpoint Selection.

Index Terms: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques; J.3 [Life and Medical Sciences]: Medical Information Systems—

1 INTRODUCTION

In the clinical routine the available time to investigate large quantities of patient data, recorded by modern medical imaging modalities (e.g., computed tomography), is very limited. As today's medical datasets may contain several hundreds of slices, it is already very time-consuming to scroll through them. As soon as a potentially pathological area is detected on a slice it can be very helpful to visualize its three-dimensional context. The prime reason which prevents a broad usage of 3D visualizations in the clinical routine is the tedious work to set up all the needed parameters. For example, the rotation in a three-dimensional space to select a good viewpoint is often not very intuitive for people, who are not dealing with computer graphics. Also the definition of a transfer function, which shows the feature of interest and its anatomical context in an expressive way often is a time-consuming trial-and-error process. Another issue which appears by moving from 2D to 3D visualization is occlusion. Clipping planes can be defined to remove structures which occlude the feature of interest. A typical set of clipping planes allows an object-aligned removal along the x-, y-, and z-axis, as well as a near and far clipping along the view direction. Finally the zoom factor for the 3D view has to be adjusted manually.

The goal of the LiveSync interaction metaphor is to derive all the parameters which are needed to present a meaningful 3D view

---

*e-mail: kohlmann@cg.tuwien.ac.at

†e-mail: bruckner@cg.tuwien.ac.at

‡e-mail: armin.kanitsar@agfa.com

§e-mail: groeller@cg.tuwien.ac.at

with minimal interaction effort. As the radiologist is usually examining the slices with a keyboard and mouse as input devices, pointing with the mouse on a structure of interest combined with pressing a hot-key is not very intrusive. In earlier work [10] we introduced the basic concept of LiveSync. To encode the quality of a viewpoint for different input parameters, viewing spheres are deformed. A viewing sphere surrounds the entire object and all possible viewpoints are located on the surface of the sphere. The viewing direction points to the center of the sphere. This concept presented the basic building blocks to achieve the live synchronization of a 2D slice view and a volumetric view. In this paper we improve and extend the previously presented techniques with the following contributions:

Sphere parameterization: An efficient sphere parameterization is used to encode the viewpoint quality for the input parameters. With a parameterization in polar coordinates, the distances between neighboring points vary a lot, depending on their distance to the poles. A multi-resolution approach which samples the sphere with uniformly distributed points is used. This allows hierarchical refinement of the sampling for efficient calculations of the overall viewpoint quality. Beside its efficiency regarding memory consumption and performance, it can be employed to find a good viewpoint that is surrounded by other viewpoints which are estimated as good ones.

Feature extraction: The extraction of a feature of interest is a crucial part for finding good viewpoints. Our region growing based segmentation automatically determines the size of the region which is necessary to specify the shape of the object of interest.

Transfer function: The manual setup of a transfer function is rather time-consuming and often not very intuitive. With increasing dimension of the transfer function space the definition becomes more and more complicated. For alleviation, the user can choose from a predefined set of transfer functions which are adjusted for typical examination procedures. With the knowledge about the distribution of scalar values within the feature of interest, it is possible to fine-tune a predefined transfer function. This is especially important when a structure is picked on the slice which is not visible in the 3D view with the currently defined transfer function.

Interaction modes: An informal evaluation indicated that a single click on the slice often results in a very good view on the interesting object in the 3D view. However, the size of the area of interest is hard to depict without further user interaction. As it is preferred to keep the user effort as low as possible, the time of keeping a hot-key pressed is taken to interactively increase the area of interest. The view is updated in short time intervals and the user can decide when an intended view is reached. In this mode a mask with the current segmentation result can be displayed to inspect the region growing process. An automatically generated history helps the user to restore and to keep previously generated views. This is especially helpful to review the intermediate states which are generated in this new interaction mode.

Feature-driven clipping: In our previous paper [10] only the setup of a view-aligned clipping plane was supported. The performed evaluation showed that often object-aligned clipping planes are preferred, especially if there is further interaction necessary to manually fine-tune the view. In the presented approach the user can choose if the view-aligned or the object-aligned clipping planes are set automatically to remove occluding structures, while preserving as much contextual information as possible. To increase the degree of preservation, importance-driven clipping techniques are integrated.

This paper is structured as follows: In Section 2, the relevant previous work is discussed. An overview of the LiveSync workflow and functionalities is given in Section 3. Section 4 describes the parameterization of the spheres. In Section 5, it is first shown how the extent of the region growing-based segmentation and the other growing parameters are derived automatically. In the following, an approach is presented to fine-tune a predefined transfer function by taking the distribution of scalar values within the feature of interest into account. Section 6 introduces the different interaction modes and Section 7 describes the feature-driven clipping. Section 8 gives numbers about the performance of the live synchronization and presents qualitative feedback from users. Finally, Section 9 concludes the paper.

2 RELATED WORK

A major task of LiveSync is the computation of viewpoint quality. The selection of a good viewpoint is a well-investigated research area for polygonal scenes. Only recently research on viewpoint selection for volumetric data originated. Kohlmann et al. [10] presented the first attempt to combine optimal viewpoint estimation and synchronized views for the visualization of medical volume data. Mühler et al. [13] presented an interesting approach for the generation of animations for collaborative intervention planning and surgical education. For the generation of these animations the selection of good viewpoints based on polygonal data in a medical context is crucial.

In the scope of volumetric data Bordoloi and Shen [2] introduced an approach based on entropy known from information theory. With their technique it is possible to determine a minimal set of representative views for a given scene. For the viewpoint selection process they take the distribution of the underlying data, the transfer function, and the visibility of voxels into account. Takahashi et al. [16] proposed a feature-driven approach for the detection of an optimal viewpoint. As a first step they define locally optimal viewpoints for identified feature components. Taking these local viewpoints into account, they compute a viewpoint which fulfills global viewpoint quality criteria. An importance-driven approach to focus on objects of interest within segmented volume data is presented by Viola et al. [19]. An expressive view on the user-selected object is generated automatically by their system. Chan et al. [5] focused on the viewpoint selection for angiographic volume data sets. They define view descriptors for visibility, coverage, and self-occlusion of the vessels to determine a globally optimal view. The final view is selected by a search over a viewpoint solution space.

To define a volume of interest (VOI) in volumetric data several approaches were presented within the last years. Tory and Swindells [17] presented ExoVis for the definition of cutouts which can be displayed with different rendering styles or transfer functions. A sketching-based approach where the user defines the interesting region by painting 2D strokes along the contour of the interesting structure is presented by Owada et al. [14]. The RGVis techniques introduced by Huang and Ma [7] present a 3D region growing approach to assist the user in locating and defining VOIs. They perform partial region growing to generate a transfer function, which reveals the full feature of interest.

Another work on semi-automatic generation of transfer functions was presented by Kindlmann and Durkin [9]. They assume that the boundary regions between relatively homogeneous materials are the areas of interest in the scalar volume. After generating a three-dimensional histogram using the scalar values, the first and the second directional derivatives along the gradient direction they can generate the opacity transfer function. A cluster-space approach for the classification of volume data was presented by Tzeng and Ma [18]. In a preprocessing step they transform volumetric data into a cluster space representation. This provides an intuitive user-interface which allows the user to operate in this cluster space. Rezk-Salama et al. [15] presented an approach where high-level transfer function models designed by visualization experts can be controlled by a user interface which provides semantic information. The non-expert user only has to adjust sliders for a goal-oriented setup of a suitable transfer function.

There are various approaches for the rendering and accentuation of identified features. Zhou et al. [22] use a geometric shape like a sphere to divide the volume into a focal and a context region. For rendering they combine direct volume rendering (DVR) with non-photorealistic rendering (NPR) techniques. Marchesin et al. [12] introduced locally adaptive volume rendering to enhance features. They modify the traditional rendering equation to improve the visibility of selected features independently of the defined transfer function. Huang et al. [8] presented an automatic approach to generate accurate representations of a feature of interest from segmented volume data. They construct a mesh for the boundary of the volumetric feature to enable high-quality volume rendering.

Regarding sphere parameterization, there is various research on how to distribute points evenly on the surface of a sphere. Bourke [3] presented an approach which distributes an arbitrary number of points over the surface of a sphere based on the standard physics formula for charge repulsion. Leopardi [11] worked on the partition of the unit sphere into regions of equal area and small diameter. Górski et al. [6] presented HEALPix which is a framework for high-resolution discretization and fast analysis of data distributed on the sphere. A hierarchical equal area iso-latitude pixelization produces a subdivision of the sphere where each pixel covers the same surface area.

3 LIVESYNC WORKFLOW

LiveSync aims to provide an optimal setup of the view parameters for the volumetric view with little additional user interaction. Manually specifying a good 3D view for structures detected in cross-sectional images is a time-consuming task. The user has to edit many parameters, such as camera position and clipping planes in order to get an expressive visualization. This process has to be repeated for each new structure of interest. LiveSync makes the process of diagnosis more efficient by automatically generating good 3D views for interactively picked structures. This functionality can be activated on demand by pressing a hot-key while pointing the mouse over a structure of interest on the 2D slice. Depending on the quality of the instantly generated result, the user can manually refine the view by adjusting the view parameters. This concept allows an efficient and non-intrusive integration of 2D and 3D visualizations in medical workstations.

Figure 1 gives an overview on the LiveSync workflow. Based on a picking action on a slice, a set of relevant view input parameters is extracted. In the original system these parameters are *patient orientation*, *viewpoint history*, *local shape estimation* and *visibility*. To get a unified representation of the parameters regarding viewpoint quality, a viewing sphere is deformed for each of them. Each of these deformed spheres contains information about the viewpoint quality. They are combined to encode an estimation about the overall view goodness. The combined sphere now contains information on how to set up the viewpoint for the best estimated view on the
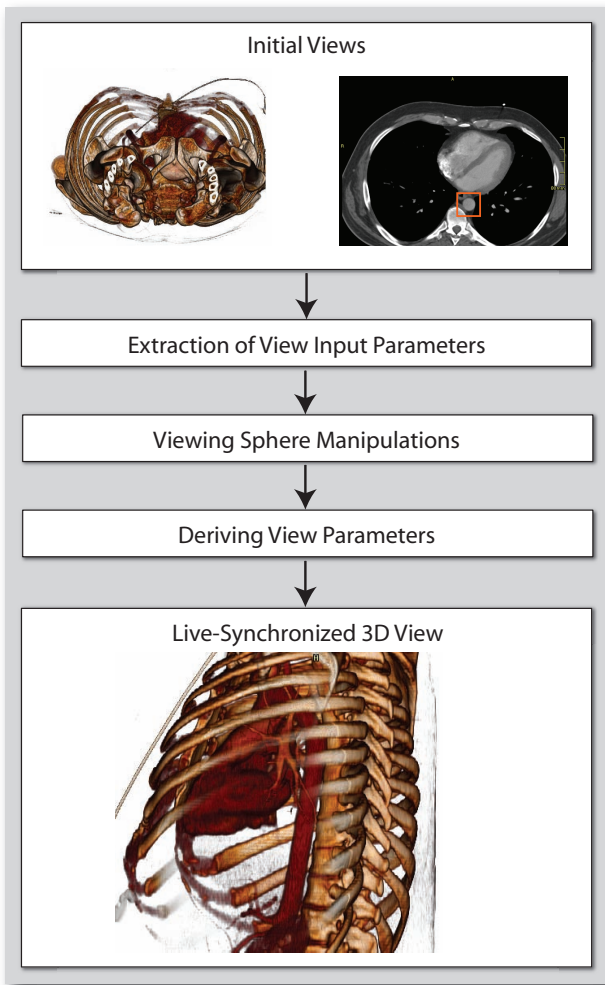
Figure 1: LiveSync workflow: The initial views are a 2D slice image and a volumetric view. A picking action on the structure of interest (surrounded by the rectangle) in the slice, starts the deformation of viewing spheres according to automatically derived view input parameters. These parameters are combined to derive the view parameters for setting up the volumetric view. As result an expressive live-synchronized 3D view is generated.

structure of interest. After the automatic setup of the clipping plane and the zoom factor, the live-synchronized view is provided.

The deformation of the spheres works as shown in Figure 2. For viewpoints which are indicated to be good ones based on one of the view input parameters, the radial distance of the surface point is increased. In Figure 2, the fully opaque eye indicates a good, the half-transparent one an average-rated, and the crossed out one a rather bad viewpoint.

The conceptual design of LiveSync is not limited to a certain number of input parameters. It can be easily extended by defining more parameters which influence the viewpoint quality. This is facilitated by the unified representation of the encoding of the viewpoint quality. Different weights can be assigned to the view input parameters to control their influence on the combined sphere. For more details on the general workflow of the LiveSync concept, we refer to previous work [10].
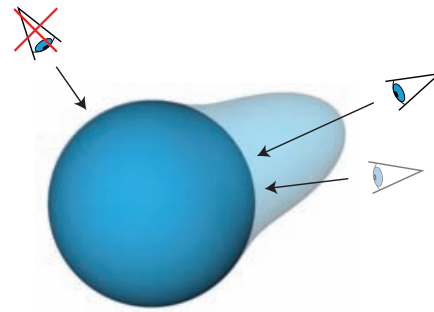

Figure 2: The quality of a viewpoint is encoded into the radial distances from a unit sphere. Good viewpoints are located at positions on the unit sphere which refer to high radial distances on the deformed sphere.

4 SPHERE PARAMETERIZATION

As the viewpoint quality of all view input parameters is encoded in a sphere parameterization, a smart way for accessing points on the surface of the sphere is important regarding performance and memory efficiency. In the original implementation there was the problem that the spheres for the view input parameters were sampled differently. Information about patient orientation, viewpoint history, and local shape information was analytically described. Due to expensive calculations, the visibility computation had to be performed in a discrete manner for a precomputed set of points on the surface of the sphere. This set of points was much smaller than for the other parameters. The computed radial distances for all spheres were stored in two-dimensional arrays with 360×180 elements via direct latitude-longitude mapping. This mapping results in a very uneven distribution of points on the surface of the sphere with much higher sampling close to the poles. Another issue has been the combination of differently sampled spheres.

4.1 Visibility Calculations

For a better understanding of the need for evenly distributed points, this section will provide a brief review of the performed calculations to generate the visibility viewing sphere. Good visibility is given if a ray from the picked point to the viewpoint exits the object of interest within few steps and if the distance until the structure of interest is occluded by other structures is high. In this case, there is high flexibility for positioning a clipping plane to remove occluding structures while preserving important anatomical context around the picked object.

Figure 3 illustrates how rays are cast from the picked point on the structure of interest to uniformly distributed points on the sphere. It has to be detected at which distance the ray exits the structure of interest. This is done by analyzing the distribution of scalar values and gradient magnitudes to decide if a voxel belongs to the structure. After a ray exits the structure of interest, opacities depending on the underlying transfer function are accumulated to detect occluding structures. As soon as a small opacity threshold is reached, the computation for a ray is terminated.

4.2 Sphere Partitioning

To achieve interactive performance the visibility calculations cannot be performed for 360×180 points. However, it is inaccurate to sample only a small number of points and to combine the sparsely sampled sphere with other, higher-sampled spheres. Moreover, taking the point with the highest radial distance after the sphere combination and before filtering, does not always lead to satisfying results. The structure of interest may be only visible through a small keyhole, and its larger part is hidden by occluding structures. In the ideal case it is very helpful for further interactions with the 3D view
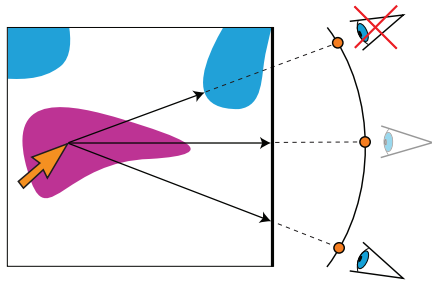
Figure 3: Starting from the picked position, visibility rays are cast to points which are equally distributed on the surface of the viewing sphere. Samples along the rays are analyzed to detect (a) when it exits the structure of interest and (b) at which position the object gets occluded by other anatomical structures.

if the provided viewpoint offers a good *view stability*. This criterion defined by Bordoloi and Shen [2] describes the maximum change in a certain view caused by small shifts of the camera position.

The capabilities of the HEALPix package [6] offer good properties for the existing needs if the techniques are applied in an elaborate way. Figure 4 (top left) shows the HEALPix base partitioning of the surface of a sphere into 12 equally sized areas with dots indicating their center positions. In each subdivision step a partition is divided into four new partitions. A nested indexing scheme can be utilized for the fast access of an arbitrary surface point at different partitioning resolutions.
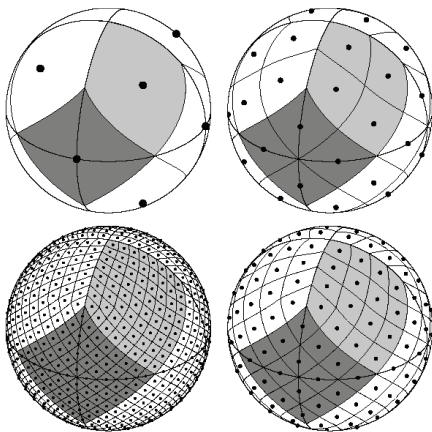

Figure 4: In its base partitioning a HEALPix sphere is divided into 12 equally sized areas. In each subdivision step an area is further divided into 4 areas (image courtesy of [6]).

Applied to the requirements of the LiveSync viewpoint quality encoding strategy, the whole sphere is sampled at an initial resolution for all view input parameters. Following, a partition of the sphere, holding the samples which indicate the best viewpoint quality is identified. This is done by summing up the radial distances of the sample positions for each partition. To achieve higher resolution, this peak partition is sampled more densely. The process of identifying a peak sub-partition with a successive hierarchical refinement can be repeated if higher resolution is required. To provide good view stability, a final filtering of the points in the target area is performed. The filtered peak is provided as the estimated best viewpoint regarding the view input parameters.

In the presented approach an initial resolution of 3076 points over the whole sphere is chosen, which corresponds to a 3.66 degrees angular distance between neighboring points. After hierarchical refinement, a peak area with 1024 sample points (0.55 degrees

angular distance) is identified and low-pass filtering is performed. This approach allows high flexibility concerning the sampling of the sphere, to detect a partition which contains a collection of many good viewpoints.

5 FEATURE-DRIVEN TRANSFER FUNCTION TUNING

A critical point in the LiveSync workflow is the definition or the extraction of the feature of interest. The shape of the feature is important to determine a good viewpoint. For example, if a blood vessel is picked, the user should be provided with a view which shows the course of the vessel and does not cut through it. This section first describes how the parameters for the region growing are controlled, before focusing on how the transfer function can be tuned with knowledge about the extracted feature.

5.1 Feature Extraction

A natural choice to segment the object around the picked point is region growing. The picked point is transformed to a voxel in 3D as seed position. Huang and Ma [7] presented with RGVis a cost function which determines if a visited voxel during the region growing process is a member of the region or not. In this approach the neighborhood of the seed point is analyzed regarding scalar value and gradient magnitude distributions to initiate the parameters of the cost function. If the seed is located close to a boundary, the growing captures the boundary of the object, whereas a seed point within a homogeneous area results in a more compact growing process.

Figure 5: The optional rendering of the segmentation mask enhances the structure of interest.

In previous work [10] the growing process was limited to a fixed region of $32 \times 32 \times 32$ voxels. In the presented approach the expansion is influenced by the spatial distribution of the points marked as region members so far. Growing progresses until the object-oriented bounding box (OOBB) of the included voxels reaches a certain limit. During the growing process the OOBB is updated at variable intervals to estimate how many more voxels may be added until the limit is reached. This strategy is superior to controlling the growing by the number of detected member voxels. The behavior of the growing process regarding the spreading of the points depends on various aspects. If just the boundary of a structure is segmented, a much lower number of voxels is sufficient to estimate
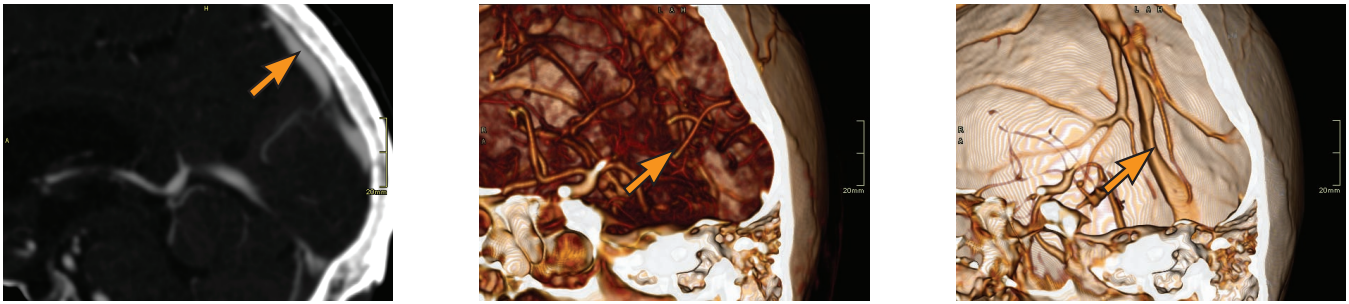
Figure 6: Left: The position of the picking on the slice. Middle: The structure of interest is hidden to a large extent by blood vessels. Right: Allowing LiveSync to fine-tune the transfer function automatically leads to an unoccluded view of the sinus vein.
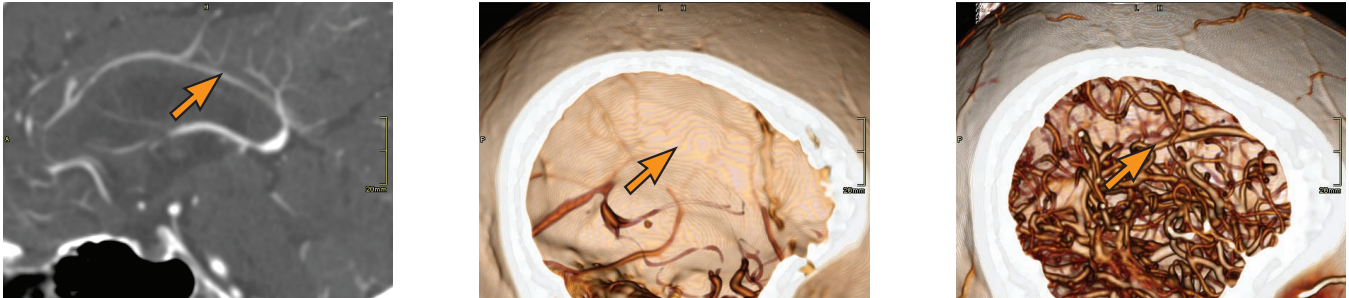

Figure 7: Left: The position of the picking on the slice. Middle: The structure of interest is not visible with the current setting of the opacity transfer function. Right: Allowing LiveSync to fine-tune the transfer function assigns opacity to the small vessels to make them visible.

Figure 8: Left: The position of the picking on the slice. Middle: Initial view with a default transfer function. Right: LiveSync-adjusted transfer function which enables a good view on the metatarsal.

the feature shape, than if the growing is performed in a very homogeneous area. Also with thin structures like blood vessels a small number of object voxels can already indicate the shape, whereas more voxels are needed for more compact areas.

As soon as the region growing stops, a principal component analysis (PCA) is performed on the member voxels to extract the three feature vectors and the corresponding eigenvalues. A metric of Westin et al. [21] is used to measure the local shape of the segmented feature from the relation between the eigenvalues. This metric allows to classify if a structure has an isotropic, a planar or linear shape. By labeling the segmented voxels with a segmentation mask, this information can be displayed in the live-synchronized volumetric view to highlight the structure of interest for a clear visual separation from its context. Figure 5 shows this enhanced visualization as result of a picking on the metatarsal in the slice view.

5.2 Transfer Function Tuning

It is frequently stated, that the definition of a transfer function to assign color and opacity to the data values, is a time-consuming and not very intuitive task. Even for the developer of a transfer function editor it often ends in a tedious trial-and-error process to generate the desired result. Usually, the necessary effort increases with the dimensions of the transfer function.

LiveSync aims to generate the 3D visualization without any additional interaction than the picking. Even the control of sliders for the transfer function design as presented by Rezk-Salama et al. [15] might lead to an unwanted distraction in the diagnosis process. The presented system allows the user to choose from a set of predefined transfer function templates, which are very well-tailored for different types of examinations. The transfer function is defined by a color look-up table and a simple ramp which assigns zero opacity to scalar values from zero to the start of the slope, increasing opacity along the slope, and full opacity from the peak of the slope to the end of the scalar range.

Inspired by the definition of a transfer function based on partial region growing in the work by Huang and Ma [7], knowledge about the distribution of scalar values within the extracted object is utilized to fine-tune an existing transfer function. This feature can be activated on demand and is especially helpful if the structure of interest is occluded to a large extent by densely surrounding objects. It is also helpful if a structure selected on the slice is hardly visible or not visibility at all applying the current opacity transfer func-
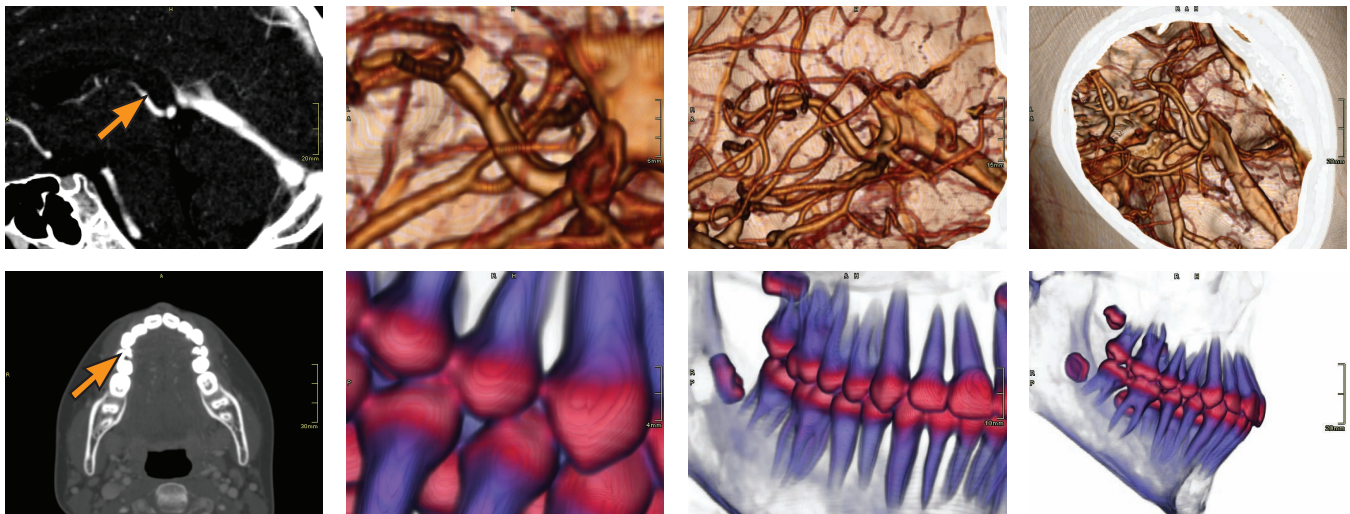
Figure 9: The slice images highlight the picking positions. Initiated by these pickings the 3D views are live synchronized at equal time intervals. Images are provided for increasing OOBB diagonal lengths from left to right.

tion. The adjustment is based on the mean value and the standard deviation of the feature's scalar values which were collected by the region growing. The center of the slope is set to the mean value and the width is adjusted to three times the standard deviation.

Figure 6 shows how the automatic setup of the ramp adjusts the transfer function. The structure of interest is visible to a much larger extent by making the surrounding blood vessels transparent. The opposite case is shown in Figure 7 as the current transfer function setup does not assign enough opacity to the smaller blood vessels. By moving the center and changing the width of the ramp which defines the opacity transfer function, the object of interest is clearly visible in the volumetric view. Figure 8 shows the result of automatic transfer function adjustment to provide a good view on the metatarsal.

6 INTERACTION MODES

The primary conceptual goal for the design of LiveSync was to interactively offer a synchronized 3D view. This shall be achieved without the need to manually adjust all the parameters to generate an expressive view. Therefore a non-intrusive user interaction technique has to be implemented which lets the user focus on his primary analysis goal. In this section, the value and the limitation of the LiveSync picking interaction will be described and extended to integrate more knowledge about the intentions of the user.

6.1 LiveSync Mode

The evaluation with a radiology technician in previous work [10] showed that the intuitive picking was sufficient to generate good results. One comment was that the handling and the mouse-over/hot-key interaction is very intuitive. However, it is rather difficult to derive in which part of an anatomical structure the user is interested. If the object is very small like a lung nodule or a polyp in the colon, the user will be provided with a view which shows the whole feature within its three-dimensional context. If the volumetric extent of the feature is rather large, like e.g., the sinus vein, then the user might either be interested in getting a close-up view of the structure around the picked point, or a view which shows the whole vein is preferable. The current zoom factor of the slice view controls the zoom factor of the volumetric view, as it indicates the size of the anatomical structure the user is interested in. To integrate more knowledge about the size of the structure, the goal was to extend

the functionality of LiveSync without introducing new interaction methods to the user.

6.2 LiveSync++ Growing Mode

Taking into consideration that region growing is performed to segment the structure of interest, the natural choice was to let the user control the size of the area of interest by keeping the hot-key pressed. This strategy allows to control the volumetric extent of the region growing. As mentioned in Section 5.1, OOBBs are computed during the growing process to estimate the extent of the initial growing. This is used to derive a reliable estimation of the local feature shape. By keeping the hot-key pressed, the region keeps on growing and the 3D view is updated in fixed time intervals. Within each interval, the growing process is continued until the diagonal length of the OOBB of the so far segmented set of voxels reaches the next predefined target size.

Integrating the knowledge about the user-indicated size of the area of interest, this can be utilized to improve the live-synchronized volumetric view regarding the demands of the user. After each growing step, the viewpoint is optimized, the clipping planes are adjusted and the zoom for the 3D view is changed. Figure 9 shows intermediate steps of the growing process. These views are provided to the users while they keep the hot-key pressed. Whenever a satisfying result is reached a release of the key stops the update of the volumetric view. There is a smooth zoom out provided for each step as the user wants to examine a larger area of interest. The zoom factor is computed by projecting the OOBB of the currently segmented set of voxels on the screen, and allowing it to cover a certain amount of the available display area, e.g., 50%.

The growing mode generates live-synchronized views at short update intervals. The user releases the hot-key or continues with the mouse movement to stop the update process. To prevent that the users miss and lose a view they like, a history mode is implemented. All live-synchronized views, including the intermediate views, are stored automatically in a view history. This is achieved by storing the adjusted parameters for the current view. The user can select a previous view from a list and store the best ones permanently.

7 FEATURE-DRIVEN CLIPPING

Viewpoint selection is critical for generating the live-synchronized 3D view. In most cases interesting structures are surrounded by other tissues. These can be removed by the definition of an adjusted

Graphics **Interface** 2008

transfer function or, more straight-forwardly, by the definition of a clip geometry. In this section, first the automatic setup of conventional clipping planes is explained before focusing on clipping planes which enable a higher degree of context preservation.

7.1 LiveSync Clipping Strategies

Typically, a medical workstation offers view-aligned and object-aligned clipping planes to interact with. As LiveSync aims to minimize the user's effort, the clipping is performed automatically to provide an unoccluded view on the structure of interest. Based on the visibility calculations described in Section 4.1, a position in the volume to place the clipping planes is computed. The plane is placed along the visibility ray where the structure of interest is not yet occluded by other structures. In our previous work [10] only view-aligned clipping planes were supported. A plane which is orthogonal to the viewing direction is moved to the clipping position and the raycasting process starts where this plane intersects the volume. This strategy provides a good view on the structure of interest. Problems might occur if the user wants to perform further interactions with the 3D view after the live-synchronized view is generated. In the worst case, the structure of interest is clipped away by manual changes of the viewpoint.

Object-aligned clipping planes are moved along the three main patient axes. The six available clipping planes can clip from left, right, anterior, posterior, head and feet. To decide which of these planes has to be set to allow an unoccluded view on the structure of interest, the plane which is most perpendicular to the viewing direction has to be identified. Then this plane is placed at the appropriate clipping position along the visibility ray. An object-aligned clipping plane remains at its position when the viewpoint is changed manually and cannot clip away the structure of interest unintentionally.

7.2 LiveSync++ Smooth Importance-Driven Clipping

The drawback of conventional clipping strategies is that they often remove anatomical context which does not occlude the object of interest. Viola et al. [20] introduced importance-driven volume rendering to remedy this problem. Their strategy uses segmentation information to generate a clipping region which follows the shape of the focus object. This approach relies on a complete segmentation of the structure in question. As our method relies on incomplete information in order to permit parameter-less interaction, the method presented by Viola et al. can lead to misleading results as a full segmentation is not available.

To remedy this problem, while still providing more context information for the picked structure, we propose an extension of the context-preserving rendering model of Bruckner et al. [4]. The advantage of this approach is that it still follows the semantics of clipping planes, but retains salient features which are important for orientation.

We compute the opacity α at each sample point along a viewing ray in the following way:

$$\alpha = \begin{cases} \alpha_{tf} \left(|g| \left(1 - |n \cdot v| \right) \right)^{\left(1 - (d-c) \right)^{\kappa}} & \text{if outside picked region} \\ \alpha_{tf} & \text{otherwise.} \end{cases}$$

where α_{tf} is the opacity as specified in the transfer function, $|g|$ is the gradient magnitude, n is the normalized gradient vector $(g/|g|)$, v is the view vector, d is the distance along the viewing ray, c is the location of the view-aligned clipping plane, and κ is a parameter which controls the transition between clipped and unclipped regions. The range of d and c is [0..1], where zero corresponds to the sample position closest to the eye point and one corresponds to the sample position farthest from the eye point. Opacity is only modulated in regions outside the focus structure identified by our local segmentation procedure. The adjustment is based on the
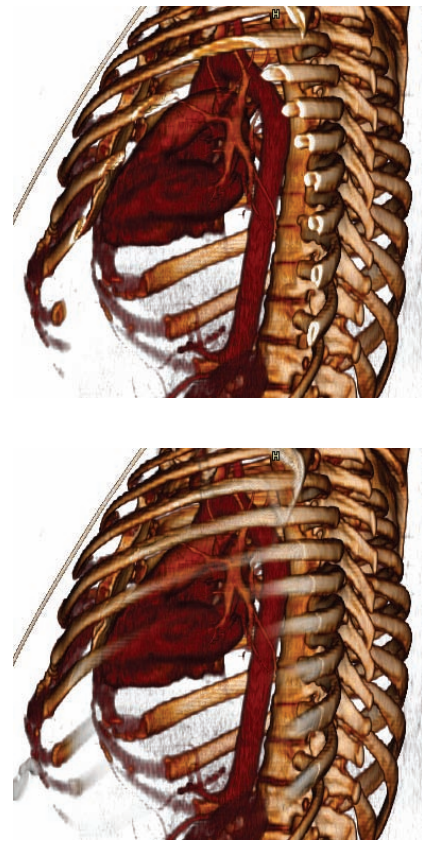

Figure 10: Top: Conventional clipping leads to a hard cut through the ribs. Bottom: Smooth clipping preserves anatomical context by reducing the opacity of the ribs which are not occluding important parts of the aorta.

idea of smooth clipping: Instead of having a sharp cut, it smoothly fades out structures that would be clipped otherwise. As the gradient magnitude indicates the "surfaceness" of a sample, its inclusion causes homogeneous regions to be suppressed. Additionally, since contours give a good indication of the shape of a three-dimensional structure, while being visually sparse (i.e., they result in less occlusion), the gradient magnitude is multiplied by $(1 - |n \cdot v|)$. The parameter κ controls the transition between clipped and unclipped regions. In all our experiments, a value of 16 has proven to be most effective. Thus, this parameter does not require user adjustment and therefore does not introduce additional complexity.

While we already achieve satisfactory results using this context-preserving clipping model, pathological cases might still cause an occlusion of the picked structure. Therefore, we additionally reduce the previously accumulated color and opacity along a ray on its first intersection with the picked region by a factor of $0.5 \left(1 - \alpha_{tf} \right)$. This lets the picked region always shine through – similar to ghosted views commonly found in medical illustrations – even if it is occluded by other structures. An example of this clipping technique is shown in Figure 10. The picking was performed on the aorta in the slice view. In the top image there is a hard cut through the ribs, whereas the smooth clipping shows the ribs which are not occluding the interesting part of the aorta with decreased opacity.

8 PERFORMANCE AND QUALITATIVE RESULTS

All computations which are done to derive the parameters to set up a live-synchronized 3D view can be performed interactively. The performance was measured on a PC configured with an

AMD Athlon 64 Dual Core Processor 4400+ and 2 GB of main memory. Compared to previous work [10] where the region growing was fixed to a limit of $32 \times 32 \times 32$ voxels, the performance is more strongly influenced by the growing process. The size of the growing area is calculated adaptively. In the vast majority of the cases it is possible to stop the growing process much earlier than with a fixed growing area and the estimation of the local feature shape is more reliable.

Besides offering better view stability, the efficient and flexible sphere parameterization helps to eliminate the need for precomputing uniformly distributed points on the sphere, to reduce memory consumption, and to increase the overall performance. The LiveSync-related computations are averaging between 50 ms and 100 ms by measuring with different data sets and picking on various structures. In growing mode as described in Section 6.2, the time to compute the intermediate views is in the same range as for the instant updates.

In previous work [10] an informal evaluation with an experienced radiology assistant has been performed. The feedback about the handling of the picking interaction and the live-synchronized views was very positive. The extensions presented in this paper are designed based on comments for possible improvements. As the LiveSync feature is integrated into a real world medical workstation which is under development by our collaborating company partner, its functionality is frequently presented in demonstrations to radiologists. Their feedback is very positive as well, and they confirm the high practical value of this feature.

9 CONCLUSION

Live synchronization of the volumetric view from non-intrusive interaction with a slice view, is a powerful concept with the potential to improve the efficiency of diagnosis in clinical routine. In this paper we presented several enhancements of the existing concept. Our goal was to integrate more knowledge about the anatomical structure the user is interested in to provide an expressive 3D visualization. As the key of the LiveSync concept is to keep the user interaction as small as possible, we did not want to increase the interaction effort or to introduce new interaction techniques.

A more efficient parameterization for the derived view input parameters was integrated to allow a hierarchical refinement of the search space for the best estimated viewpoint. This parameterization is utilized to provide a view with a high degree of view stability. An adaptive region growing was presented that allows to extract the part of the feature which is needed to estimate its local shape in a reliable way. Information about the scalar value distribution within the feature of interest is used to fine-tune the existing opacity transfer function. This helps to enhance the visibility of the object the user is interested in. The growing interaction mode integrates knowledge about the volumetric extent of the area of interest without increasing the user interaction. Especially an appropriate zoom factor for the 3D view can be derived from this information. A new clipping technique was integrated to increase the preservation of valuable anatomical context while guaranteeing an unoccluded view on the structure of interest.

REFERENCES

[1] OsiriX Medical Imaging Software. Available online at http://www.osirix-viewer.com, December 2007.

[2] U. D. Bordoloi and H.-W. Shen. View selection for volume rendering. In *Proceedings of IEEE Visualization 2005*, pages 487–494, 2005.

[3] P. Bourke. Distributing Points on a Sphere. Available online at http://local.wasp.uwa.edu.au/∼pbourke/geometry/spherepoints/, December 2007.

[4] S. Bruckner, S. Grimm, A. Kanitsar, and M. E. Gröller. Illustrative context-preserving exploration of volume data. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1559–1569, 2006.

[5] M.-Y. Chan, H. Qu, Y. Wu, and H. Zhou. Viewpoint selection for angiographic volume. In *Proceedings of the Second International Symposium on Visual Computing 2006*, pages 528–537, 2006.

[6] K. M. Górski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622:759–771, 2005.

[7] R. Huang and K.-L. Ma. RGVis: Region growing based techniques for volume visualization. In *Proceedings of the 11th Pacific Conference on Computer Graphics and Applications*, pages 355–363, 2003.

[8] R. Huang, H. Yu, K.-L. Ma, and O. Staadt. Automatic feature modeling techniques for volume segmentation applications. In *Proceedings of IEEE/Eurographics International Symposium on Volume Graphics 2007*, pages 99–106, 2007.

[9] G. Kindlmann and J. W. Durkin. Semi-automatic generation of transfer functions for direct volume rendering. In *Proceedings of the IEEE Symposium on Volume Visualization 1998*, pages 79–86, 1998.

[10] P. Kohlmann, S. Bruckner, A. Kanitsar, and M. E. Gröller. LiveSync: Deformed viewing spheres for knowledge-based navigation. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1544–1551, 2007.

[11] P. Leopardi. A partition of the unit sphere into regions of equal area and small diameter. *Electronic Transactions on Numerical Analysis*, 25:309–327, 2006.

[12] S. Marchesin, J.-M. Dischler, and C. Mongenet. Feature enhancement using locally adaptive volume rendering. In *Proceedings of IEEE/Eurographics International Symposium on Volume Graphics 2007*, pages 41–48, 2007.

[13] K. Mühler, M. Neugebauer, C. Tietjen, and B. Preim. Viewpoint selection for intervention planning. In *Proceedings of Eurographics/IEEE VGTC Symposium on Visualization 2007*, pages 267–274, 2007.

[14] S. Owada, F. Nielsen, and T. Igarashi. Volume catcher. In *Proceedings of ACM Symposium on Interactive 3D Graphics and Games 2005*, pages 111–116, 2005.

[15] C. Rezk-Salama, M. Keller, and P. Kohlmann. High-level user interfaces for transfer function design with semantics. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1021–1028, 2006.

[16] S. Takahashi, I. Fujishiro, Y. Takeshima, and T. Nishita. A feature-driven approach to locating optimal viewpoints for volume visualization. In *Proceedings of IEEE Visualization 2005*, pages 495–502, 2005.

[17] M. Tory and C. Swindells. Comparing ExoVis, orientation icon, and in-place 3D visualization techniques. In *Proceedings of Graphics Interface 2003*, pages 57–64, 2003.

[18] F.-Y. Tzeng and K.-L. Ma. A cluster-space visual interface for arbitrary dimensional classification of volume data. In *Proceedings of Joint Eurographics - IEEE TCVG Symposium on Visualization 2004*, pages 17–24, 2004.

[19] I. Viola, M. Feixas, M. Sbert, and M. E. Gröller. Importance-driven focus of attention. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):933–940, 2006.

[20] I. Viola, A. Kanitsar, and M. E. Gröller. Importance-driven feature enhancement in volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):408–418, 2005.

[21] C.-F. Westin, A. Bhalerao, H. Knutsson, and R. Kikinis. Using local 3D structure for segmentation of bone from computer tomography images. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition 1997*, pages 794–800, 1997.

[22] J. Zhou, M. Hinz, and K. D. Tönnies. Focal region-guided feature-based volume rendering. In *Proceedings of the 1st International Symposium on 3D Data Processing, Visualization, and Transmission 2002*, pages 87–90, 2002.