Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

ALEXANDER WIEBEL, FRANS M. VOS UND
HANS-CHRISTIAN HEGE

Perception-Oriented Picking of Structures in Direct Volumetric Renderings

ZIB-Report 11-45 (Dezember 2011)

Herausgegeben vom
Konrad-Zuse-Zentrum für Informationstechnik Berlin
Takustraße 7
D-14195 Berlin-Dahlem

Telefon: 030-84185-0
Telefax: 030-84185-125

e-mail: bibliothek@zib.de
URL: http://www.zib.de

ZIB-Report (Print) ISSN 1438-0064
ZIB-Report (Internet) ISSN 2192-7782

Perception-Oriented Picking of Structures in Direct Volumetric Renderings

Alexander Wiebel* [1], Frans M. Vos [2] and Hans-Christian Hege* [1]
[1]Zuse Institute Berlin (ZIB), Germany [2]TU Delft & AMC Amsterdam, Netherlands

2011-12-19

Abstract

Radiologists from all application areas are trained to read slice-based visualizations of 3D medical image data. Despite the numerous examples of sophisticated three-dimensional renderings, especially all variants of direct volume rendering, such methods are often considered not very useful by radiologists who prefer slice-based visualization. Just recently there have been attempts to bridge this gap between 2D and 3D renderings. These attempts include specialized techniques for volume picking that result in repositioning slices.

In this paper, we present a new volume picking technique that, in contrast to previous work, does not require pre-segmented data or metadata. The positions picked by our method are solely based on the data itself, the transfer function and, most importantly, on the way the volumetric rendering is perceived by viewers. To demonstrate the usefulness of the proposed method we apply it for automatically repositioning slices in an abdominal MRI scan, a data set from a flow simulation and a number of other volumetric scalar fields. Furthermore we discuss how the method can be implemented in combination with various different volumetric rendering techniques.

1 Introduction

Direct volume rendering (DVR) [Sab88] is the state-of-the-art for the display of volumetric data from medicine, engineering and the sciences. As a flexible and versatile tool, it is adaptable to virtually all application problems dealing with 3D scalar fields. The latest developments of the power of computer hardware allow DVR to be used interactively even on consumer type systems. Although this makes it available for the analysis and inspection of data from medical imaging devices, the radiologists responsible for these tasks still mainly rely on the examination of slice-like depictions (including multi-planar reformatting, MPR). Motivated by this fact previous work has already addressed the combination of DVR and MPR representations [KBKG07, KBKG08, KBKG09, Vli08]. Providing interaction techniques (commonly called volume picking, point picking or volume pinpointing) that allow to pick in the volumetric rendering to adjust a slice, and vice versa to pick on the slice to reorient the DVR, make it possible to integrate DVR in the daily routine of the radiologists. DVR can serve as overview while the slices are still used for the detailed examination the radiologists need to perform.

Keeping this background in mind, the motivation for the technique presented in this paper is threefold: current methods either use metadata or are designed for medical data only, or provide only very basic picking techniques like *first-hit* or opacity-threshold. The aim of this paper is to introduce a picking technique that circumvents all these disadvantages by taking a perception-based view. Picking is probably the most intuitive interaction technique possible because it is the technical equivalent to one of the most natural actions in the real world: *pointing at something*. We present a method that enables users to intuitively select spatial positions in volumetric renderings.

*wiebel@zib.de
*hege@zib.de

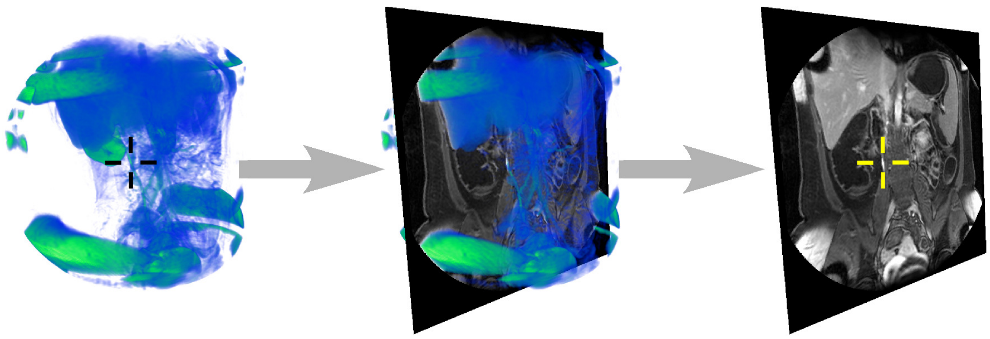

Figure 1: Perception-oriented picking in DVR applied to abdominal magnetic resonance imaging (MRI) scan. The picked position is used for selecting a slice depicting the picked feature (vessel). The position is marked with crosshairs in the left and right image.

In particular, the main contributions of this paper are

- a short summary of how picking is realized for different rendering techniques working on volumetric data,
- a new, perception-oriented technique allowing to pick positions in arbitrary direct volume rendering images,
- the method's independence of any information apart from the volume data and the transfer function of the direct volume rendering,
- its applicability to renderings for any volumetric scalar data set and any types of transfer functions (e.g also "foggy" looking images),
- and its usefulness for navigating (e.g. selecting slices) in the resulting visualizations.

Thereby, we intend to pave the way for further application of DVR in application areas that still are reluctant to adopt this fundamental visualization technique

2 Related Work

In this section we review the previous work on picking in volumetric renderings and the combination of direct volume rendering with slices. Concerning perception there has also been a lot of work in visualization research, see e.g. [TM04, HE11]. However, this work is mainly concerned with designing visualization according to perception principles, and rather than how volumetric visualizations are perceived.

2.1 Picking

Direct volume rendering, has been around for over twenty years now [Sab88] and over time has developed into an interactively usable rendering technique (see e.g. [KW03, AGI*08]), which has resulted in research that aims at facilitating the interaction with volumetric depictions. Volume picking, the interaction technique that is in the focus of this article, has been adapted from its well-known predecessor that is used for picking real geometry like surfaces. Accordingly, the first volume picking techniques mimicked the surface picking by searching for the first surface-like structure along the viewing ray passing through the picked screen position. Gobbetti et al. [GPZT98] introduced the most widely used technique. It searches along the ray using the usual compositing scheme (described in Section 4) and stops as soon as the accumulated opacity exceeds a user defined threshold. This means a surface is assumed to be at locations where the opacity threshold is exceeded. The endpoint of the search is returned as 3D position resulting from the picking. A simplified version of this approach sets the threshold to zero. This results in selecting the first not completely transparent position in the volume (*first-hit*).

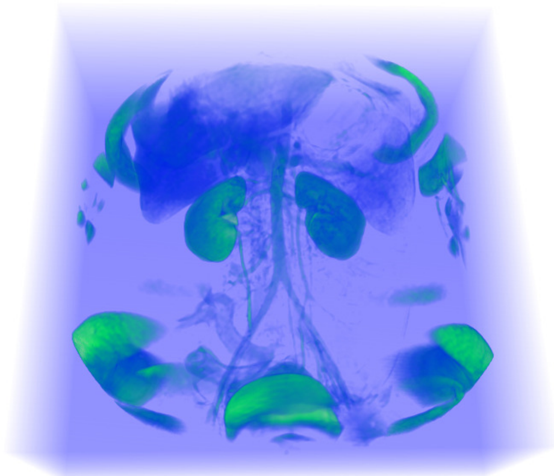

Figure 2: Problem of *first-hit* method with zero threshold and "foggy" rendered image. The resulting position will be on the bounding box instead of, e.g., the kidneys because every position in the volume has non-zero opacity.

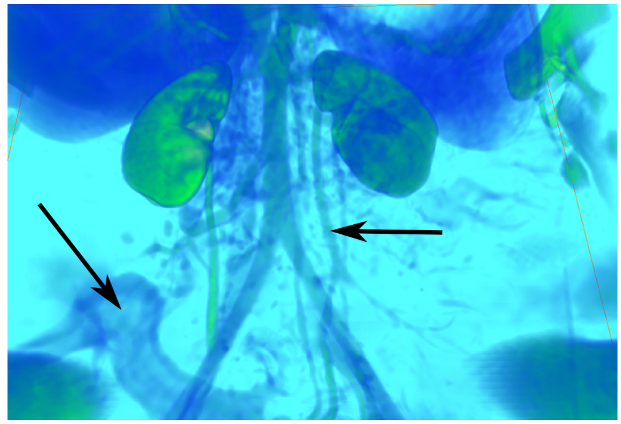

Figure 3: Problem of threshold picking method with relatively transparent regions. Such regions might be missed although they are clearly visible. Example regions (vessel, terminal ileum) are marked with arrows.

Both variants can result in undesired results. Using a zero threshold will return positions in regions surrounding the features in "foggy" looking renderings (see Fig. 2). If, on the other hand, the threshold is non-zero some relatively transparent but still visible regions might be missed (see Fig 3).

Another widely used method selects the largest data value along the ray. While yielding perfect results in conjunction with maximum intensity projection renderings, this technique is not suitable for DVR in general. For common DVR it can result in selecting positions that are completely transparent, i.e. deliberately not shown, due to the selected transfer function. This and the above technique are described in more detail in Section 3. Toennies and Derz [TD97] present a technique that searches for user-defined data values or user-defined properties of metadata along the ray. In our setting, it suffers from the same problem as the previously described method. Bruckner et al. [BŠG*09] select the position along the ray that contributes most to the final pixel. They report that it works nicely with the special volume rendering technique they used in their *BrainGazer* system. As the sample contributing most does not necessarily have to belong to the most visible *object*, i.e. the group of samples contributing most to final pixel, we use a the most contributing interval in our approach.

The following list of visualization tools and their volume picking techniques gives an impression of the use of the techniques: MeVisLab [MVL] provides a technique selecting the maximum data value as well as opacity threshold-based picking. Voreen [MSRMH09] and Avizo [Avi] use the first-hit approach. VTK [SML98] and thus ParaView [Squ08] employ the opacity threshold method [NCP].

Kohlmann et al. [KBKG09] employ a more sophisticated picking method called *contextual picking* that is especially tailored to medical data in DICOM [BH92] format. It uses the meta information given in the DICOM files to deduce which anatomical parts of the volumetric image the user intends to pick (e.g. angiography→vessels). Very few, initially user specified, ray profile samples are matched against the data curve along the viewing ray to find the intended structures. As the matching identifies the approximate extent of the picked structure, Kohlmann et al. are able to provide picking positions either on the front of the structure or in its center. Malik et al. [MMG07] use ray-profiles similarly in a different context, i.e. the division of the data into different *peelable* layers. Like our method, they use the derivatives of the ray-profile to find "features" along the ray. However, they search for features in the data whereas our method searches for features in the visible rendering (profile of accumulated opacity along the ray). Additionally, in contrast to our method their *transition points* are extrema and are thus easily detected as zeros of the first derivative. Another peeling technique somewhat related to the present work is the so-called opacity peeling by Rezk-Salama and Kolb [RSK06]. Opacity peeling uses several rounds of opacity accumulation (each up to some opacity threshold) to render layers originally occluded behind other rendered layers of the data. The users can interactively select the layer they would like to see.

The most recent paper on picking in volumetric rendering we are aware of has been made by Peng et al. [PRL*10]. They use two different techniques: (1) a one-click method restricted to their data with a blob-like structure where it is easy to guess the desired position as the center of the blob hit by the viewing ray; (2) a two-click method for which the user clicks on the desired location from two different viewing directions. The picked position is then the (fuzzy) intersection of the two viewing rays. This method works for arbitrary data but the desired location has to be visible from both viewing directions. The method we present is superior to Peng's method in so far as it allows to pick in renderings of arbitrary data with only one mouse click (or similar pointing action) from one viewing direction.

2.2 Combining Slices and DVR

Many techniques dealing with isosurfaces in volumetric data provide picking on slices for reorienting the isosurface to a view point providing good visibility of the selected position. Picking on the surface is often used to change the position of a slice in the data. Just recently this has been combined with picking in volume renderings by Kohlmann et al. [KBKG08, KBKG07] and others [BKKG08, Vli08]. In their work, picking on a slice results in a reorientation of the volume rendering and a local adaption of the opacity (transfer function) such that the view on the selected position is improved. For the reverse direction, i.e. for picking in the volume and adapting the position of the slice, they use either the first-hit or their contextual picking approach that we described above. In their framework the selected position can also be used for placing labels.

3 Picking for Different 3D Rendering Methods

As picking seems to be the most natural way of interacting with renderings of three-dimensional data or objects, there are many applications where picking is readily available. In the following we give a short overview of visualization techniques for three-dimensional data and how picking is realized in their context. In section 5, we introduce *perception-oriented picking* as the core part of this paper.

Isosurfaces Contour or isosurface extraction is probably the most common 3D visualization technique applied to volumetric data. As the displayed entities are surfaces (most often consisting of triangles), selecting the position to be picked is straight forward. One simply computes the intersection of the viewing ray with all triangles and selects the intersection point that is closest to the observer. This method is exact as long as no transparency is present.

Iso Ray-Casting A variant of isosurfaces working with ray-casting is the so-called iso ray-casting [PSL*98]. Instead of extracting the surface explicitly (in most implementations) only the first position along the ray that has/exceeds the isovalue is rendered with appropriate shading. Picking will result in the mentioned position. Regarding exactness the same constraints as for normal isosurfaces are valid.

MIP Maximum intensity projection (MIP) [WMLK89] is a technique that renders the signal value for the sample with the largest value along the viewing ray. The color of the pixel thus represents only one position in the volume. Consequently, the perfect choice for the picked position is the position of the maximum data value along the viewing ray.

mIP Minimum intensity projection (mIP,MinIP) (see e.g. [PB07]), in analogy to MIP, renders the signal according to the value of the minimum along the viewing ray. Picking thus results in the position with minimal value along the ray.

AIP The pixel color of an average intensity projection (AIP) is determined by a mapping from the mean value of all values along the viewing ray [PB07]. For this and other techniques [ME04] mimicking X-ray behavior, there is no obviously intuitive position for picking in the renderings. However, as AIP is mainly applied to slabs [LHH*08] instead of complete volumes, a reasonable choice can be the center of the slab, i.e. the mean of the viewing ray's entrance and exit points regarding the slab.

CVP According to [PB07] closest vessel projection (CVP) has been introduced by Siebert et al. in [SRP91] to fix a disadvantage of MIP. While MIP shows the sample with maximum intensity thus possibly hiding less bright features of interest in front of very bright features, CVP uses a threshold and colors the pixel according to the sample first exceeding this threshold. Thus the position picked in a CVP should be the first one exceeding the intensity threshold along the viewing ray.

DVR There is no unique way to implement picking for direct volume rendering (DVR) with general transfer functions. Such a picking is the main topic of this paper and is thus covered throughout the paper. Previous work on the topic has been reviewed in Section 2.1. Just to recall, a straight forward method for picking in DVR is the so-called *first hit* approach that can result in either the first position along the ray having non-zero opacity or the position up to which the accumulated opacity has reached a certain threshold.

Transparent Surfaces Similar to the case of DVR, there is no unique way to determine the picked position for nested transparent surfaces. As DVR can produce renderings similar to transparent surfaces, we will discuss this case in conjunction with DVR. Nevertheless, note that when assuming that a user picks with special "care" a technique by Mühler et al. [MTRP10] can predict the intended surface. Their term for this "care" is "conscious pointing" [MTRP10].

Slices Slices are the standard rendering used by most radiologists. In addition to the originally measured slices (often axial), slices in other directions can be shown. The latter is frequently called multi-planar reformatting (MPR). Picking on slices is simple and exact because a slice is a simple quadrangular graphical object. Picking works like for any other surface (see above).

Other Approaches that render volumetric data in a non-3D fashion like contour trees [CSA00] or scatter plots do not lend themselves to the usual idea of picking. However, there are other specialized ways to determine the spatial location for a certain rendered pixel in the context of these examples.

The results of all rendering methods for which we described a unique picking approach are *perception-oriented* in the same sense as the *perception-oriented picking* we introduce for DVR: exactly the *seen* 3D position is selected.

4 Background

For the description of the proposed picking approach, a basic understanding of the volume rendering procedure is necessary. We therefore give a summary of the most relevant aspects and along this way we introduce the notation.

4.1 Volume Rendering Integral

As DVR tries to make volumetric data directly visible to the user, its most natural implementation is casting rays along the viewing direction through the volume and accumulating color information for the values of the volumetric data along these rays. The density of the rays and the samples along the rays are chosen to cover the volume sufficiently well. The color information for the data values is determined by the so-called *transfer function*. The mentioned accumulation can be formalized mathematically in the *volume rendering integral* [Max95, EHK*06, PB07]:

$$I(r_{max}) = I_0 e^{\int_{r_0}^{r_{max}} \tau(t)dt} + \int_{r_0}^{r_{max}} Q(s) e^{\int_{s}^{r_{max}} \tau(t)dt} ds$$

In this equation, I is the intensity in a color channel resulting from accumulating the color for a certain distance along the ray. $[r_0, r_{max}]$ is an interval along the ray, with r_{max} being at the eye point and r_0 at the back end of the volume; s is a parameter in this interval; τ is the attenuation coefficient and Q the source term describing emission for a certain sample.

For a numerical approximation the volume rendering integral has to be discretized: compositing (accumulation) is performed for a finite number of samples along the ray. The iterative computation of the discretized version in a *front-to-back* fashion can be denoted as follows [EHK*06]:

$$c_{n+1}^{acc} = c_n^{acc} + (1 - \alpha_n^{acc}) c_n^{src} \quad (1)$$

$$\alpha_{n+1}^{acc} = \alpha_n^{acc} + (1 - \alpha_n^{acc}) \alpha_n^{src} \quad (2)$$

Here, c denotes color, α denotes opacity, n denotes the step number, *acc* indicates the accumulated values and *src* indicates values of the transfer function for the data found at the current sample position.

4.2 Compositing

Equation 2 describes the steps that have to be performed to compute the opacity at a certain sample on the ray. This opacity is accumulated along the ray up to that position. It determines how much the final pixel value is influenced by the values of the samples on the ray that lie behind the current sample. Later in this paper we will be concerned with how α^{acc} varies along the ray. Therefore it is worth noting two important properties of α^{acc} that can be easily deduced from Eq. 2: first, if we assume α_0^{acc} to be zero and $\alpha_n^{src} \in [0,1]$, the accumulated opacity will never be larger than one. Second, with the same assumptions, α^{acc} is monotonically increasing along the ray. Together, this implies $\alpha_n^{acc} \in [0,1]$.

Another important fact about the compositing in Eq. 2 is that with a change of the sampling density along the ray the series of accumulated opacity changes. More samples result in a faster increasing opacity compared to the location of the samples along the ray. This can be compensated by scaling α^{src} with respect to the sample density (*opacity correction*) [EHK*06]. Opacity correction is also necessary if non-equidistant samples are used. For sake of simplicity, we restrict all explanations to equidistant samples throughout this paper. All presented methods are easily extendable to this general case.

5 Perception-Oriented Picking for Direct Volume Rendering

In this section we give a detailed description of the new *perception-oriented picking* technique. A comparison with previous techniques that emphasizes its advantages is provided in Section 6.

The overall procedure of all picking techniques is similar. First, the user clicks on a position in the screen. This position and the user's viewing direction are transformed from screen coordinates into world coordinates. The result is then used to cast a ray through the scene (see Fig. 6). Along this ray a number of samples are used to gather information about the volume data. Finally, certain criteria are applied to the gathered data to determine the position resulting from the pick. This last step is the one that the new method improves.

5.1 Perception-Oriented Picking Criterion

At the heart of the of the new technique are the characteristics of the values of α^{acc} along the viewing ray, i.e. the discretized version of the opacity accumulation described by the volume rendering integral. Previous work on volume rendering already noted that accumulated opacity along the ray is strongly correlated with the *visibility* of regions or features in the volume [WZC*11, BS05, MWCQ11, CM09, CWM*09]. We hypothesize (confirmed by our experiments) that the user most of the time *sees* those features at a screen position that have the highest opacity contribution or in other words, the highest *jump* of α^{acc} along the ray (Fig. 4). The amount of opacity contribution of a feature thus coincides with the feature's influence on the final color of the pixel, i.e. with what is visible. This means that a user's perception does not only depend on the optical properties of a single location, but on the properties of a number of consecutive locations. Furthermore, it does not depend on the steepness of the increasing opacity but on how much the opacity increases in an interval of consecutive samples. An evident example of this effect can be seen in Figure 8.

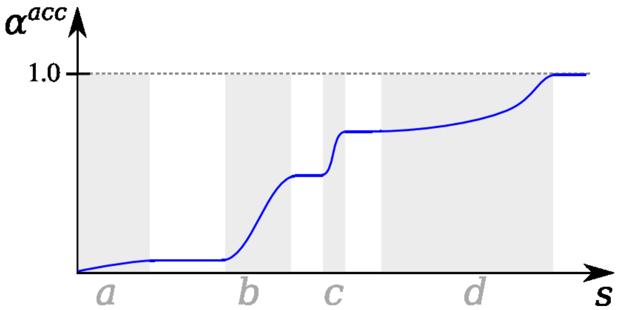

Figure 4: Jumps in accumulated opacity α^{acc}. Note that the jump denoted c is steeper than jump b but that b is higher than c. The increase in interval d represents the feature with largest extent while contributing only a small amount to the overall opacity.

Finally, considering the changes of opacity for selecting the picked position is sensible because usually high opacity is assigned to important features during transfer function design, or in short: opacity correlates to importance.

Figure 4 illustrates the criterion. Here, the largest jump can be found in region b, while the strongest ascend, i.e. the steepest jump, appears in region c. Consequently, the region used to determine the picking position is b.

To determine the highest jump, the first task is to define the regions of Figure 4 as *intervals* $I = [i_0, i_{max}] \subset [r_0, r_{max}]$ along the ray. Thereafter, the difference between α^{acc} at the start and the end of the interval, i.e. the jump j as

$$j = \alpha^{acc}(i_0) - \alpha^{acc}(i_{max}),$$

has to be computed. Extracting the boundaries i_0 and i_{max} of the jumps is similar to the task of edge detection [Mar82] in one dimension. Consequently, our method for detecting the boundaries is inspired by computer vision methods [Mar82] and incorporates the second derivative of α^{acc}. We denote the first derivative of α^{acc} as β^{acc} and the second derivative as γ^{acc}. Figure 5 illustrates the idea behind our method for extracting the interval boundaries. In principle, the boundaries are the positions where the second derivative γ^{acc} has zero crossings from below, i.e. from negative to positive values. This criterion, however, is only reliable if α^{acc} is strictly increasing. As α^{acc} has plateau-like regions and thus γ^{acc} has extended regions where it is constantly zero, the criterion is adapted as follows. The lower bounds i_0 of such intervals are the positions where accumulated opacity starts to grow stronger, that is where γ^{acc} becomes positive after being negative or zero. The criterion for the upper bounds i_{max} is that α^{acc} stops decreasing again. For γ^{acc} this means that it becomes zero or positive after being negative.

After having determined the interval boundaries and having computed all jumps j one simply has to select the interval with the largest jump j. This is the interval dominantly perceived at the picked screen position.

5.2 Front vs. Center of Perceived Feature

The criterion described above does not directly yield a position. It only yields the interval seen most prominently along the viewing ray through the picked screen position. This, however, is not a problem but rather an important characteristic of the criterion because it allows to choose the final position according to the task at hand. For labeling features in the volume rendering the front most position of the feature is of interest, whereas for repositioning slices to display most of the picked feature the center of the feature is of interest. This has also been noted by Kohlmann et al. [KBKG09] and is implemented for their contextual picking.

For the perception-oriented picking determining the center and the front position is straight forward because the front and back positions are implicitly computed as the start and end of the jump interval. A feature's front is simply the first position $\alpha^{acc}(i_0)$ of the interval corresponding to the largest jump. A feature's approximate center is the center i_c of the interval, i.e.

$$i_c = \frac{1}{2}(i_0 + i_{max}).$$

5.3 Implementation Details

In our implementation, casting the ray through the volume is realized by a combination of usual surface picking and straight forward ray casting on the CPU. We draw a completely transparent bounding cube (*proxy geometry*) around the volume rendered data in the scene. The standard geometry picking mechanism of our scene graph is then used to determine the position where the viewing ray intersects the proxy geometry and enters the data volume

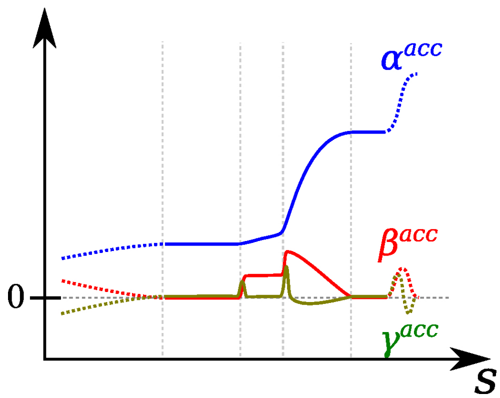

Figure 5: Detection of borders of jumps via first and second derivative of α^{acc}. The blue curve represents the accumulated opacity, the red curve its first derivative β^{acc} and the green curve its second derivative γ^{acc}. The dashed gray lines mark the detected borders. The curves are only sketched for illustration purposes and thus are only qualitatively correct.

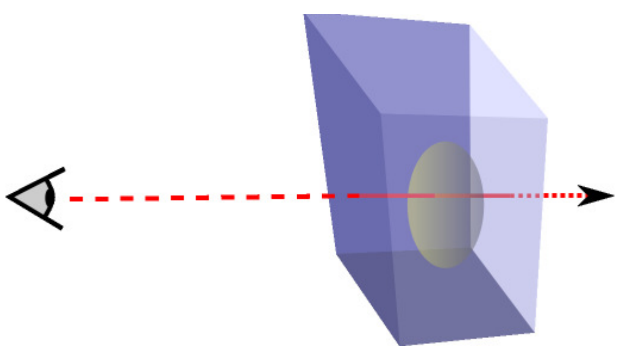

Figure 6: Ray casting for perception oriented picking in DVR. The dashed line is the part of the ray that is determined by common object picking of the proxy geometry. Samples on the solid part of the ray are used for perception-oriented picking. Dotted line is outside the bounding cube of the data set. Stepping along the ray will be stopped before reaching it.

(see Fig. 6). The direction of the ray is computed as the difference between the intersection point and the camera position or eye point. With this information we can step through the volume and gather the desired information. As soon as a step lies outside the data set's bounding box we stop gathering information. The information obtained for each step are the data value at the position and the result of applying the transfer function to this data value, i.e. color C_n^{src} and opacity α_n^{src}. Using Equation 2, these values are accumulated to provide the values of α^{acc} along the ray.

At this point it is worth noting that the parameters of the used DVR implementation and of the procedures described above need to be coordinated. The reason for this lies in the usual compositing (Eq. 2) which does not incorporate the distance of the samples. Thus, if the sample distances of DVR and picking are not equal, the accumulated opacity may vary differently along the viewing ray. The following small example demonstrates the possible issues. Consider the DVR using half the step size, i.e. twice as many samples, as the ray for the picking. Then the opacity accumulated along the ray for the DVR will increase much faster than that accumulated for the picking. If α^{acc} reaches 1.0, i.e. complete opaqueness, for the DVR it will probably not have reached 1.0 for the picking. As a result the picking will classify far away features as still visible although they can not be seen in the DVR image. This in turn can lead to picking features not visible in the rendering, which is definitely not intended.

The easiest way to achieve consistency between DVR and picking is to use the same number of steps and the same step size. If this cannot be achieved, then the previously mentioned *opacity correction* needs to be applied during compositing.

5.4 Transparent Surfaces

Figure 8, the last example that will be described in Section 6, shows that a *perception-oriented picking* for transparent surfaces can be implemented analogous to the method for DVR. One simply traces a ray through the volume, accumulates or composits the opacities of the different surfaces, and finally selects the surface that has the largest opacity contribution after compositing. Computing the derivatives of α^{acc} is not necessary (and not possible due to discontinuities) in this case because the intervals are the surfaces themselves and thus infinitely small.

6 Results, Comparison and Discussion

The aim of the presented method is to provide a picking technique that does not need any metadata, can be applied to volumetric scalar fields from all application domains and nevertheless picks the really observed 3D location corresponding to a selected 2D position. To demonstrate these characteristics we applied the method to a selection of very different volumetric scalar fields. As the most sophisticated previous picking techniques come from the area of medical visualization, an abdominal MRI scan involving intravenous contrast is our first example. Figure 1 shows how a position in a DVR image (DVRI) is picked, how a slice with the appropriate orientation is positioned so that it cuts the picked vessel, and how the slice can be subsequently used to examine the vessel in detail. The DVR is hidden in the final image to provide a completely free view on the slice. Figures 3 and 2 show DVRIs of the same data set but with different transfer functions. As adumbrated before, *first-hit picking* fails for the DVRI in Figure 3 if the threshold is chosen too high and for the DVRI in Figure 2 if the threshold is chosen too low (e.g. zero). In the first case the cast ray would go through the volume without identifying any position as picked. In the second case the ray tracing would stop as soon as it reaches the bounding box of the data set because opacity can be found everywhere. *Perception-oriented picking* can handle the DVRs of all three transfer functions correctly.

The second example data set comes from a numerical simulation of a flow around an ellipsoidal body. The images in Figure 7 show DVR of the vorticity $\|\nabla \times \mathbf{v}\|$ of the velocity vector field $\mathbf{v}$. For illustration purposes the images have been rotated so that the flow comes from below. Like for the MRI data set, the steps of the *perception-oriented picking* are shown. Additionally, the curves of the accumulated opacity α^{acc} illustrate the interval selection. While *contextual picking* is possibly applicable to the MRI data set, it is definitely not applicable for the flow field as there are no structures that can be named and matched for detection in this data set. One might think of vortices as such structures, but there is still no vortex definition commonly agreed upon.

A synthesized scalar function increasing from two locations provides the data for the third example. The transfer function used for rendering produces two balls that are visible in the DVRIs of Figure 8. The lower right image in the same figure shows the series of the accumulated opacity α^{acc} and its first (β^{acc}) and second (γ^{acc}) derivatives. The first, last and central location of the selected interval are marked by gray bars. As our picking criterion suggests, the marks coincide with zero crossings of γ^{acc}. Although the two jumps corresponding to the first two peaks of β^{acc} are steeper, the criterion selects the marked interval because it exhibits the highest jump. The result is that a position in the shell of the ball in the background is picked through two transparent shell areas of the ball in front of it. This example also shows that material boundaries parallel to the viewer, which are usually well perceived, are easily picked because they are represented by a long and strong increase in opacity and thus a high jump of accumulated opacity.

6.1 Video

The presented technique is intrinsically interactive and thus hard to demonstrate in still images. Therefore, a video with a live demonstration using the described and some additional data sets accompanies this paper.

6.2 Limitations

As may be deduced from the images throughout this paper, the presented method deals with volume rendering using the standard emission-absorption model. This does not impose any constraint on the type of transfer function (e.g. one-dimensional vs. multi-dimensional). However, we did not investigate how the method deals with images in which local illumination has been applied after evaluating the transfer function. Perception theory [Mar82] tells us that lighting, color and context influence the perception of transparency. Therefore, we expect that the method will have to be extended to correctly handle volume rendering using local illumination. Still, we will probably not have to use the most complex computer vision methods because we have much more information than only the resulting image. The data and the transfer function are highly valuable information for the picking task.

Also, our current implementation does not ensure that close positions in screen space also result in close 3D locations. In noisy data sets this might be preferable. We would explore a simple solution in conjunction with the handling of local illumination: The 3D locations corresponding to positions lying next to (probably on pixel base) the picked position can be averaged. In order to avoid "smoothing" out edges of interest, this approach will probably be restricted to positions where the most 3D locations are similar. Overall, this problem will only become relevant if we allow the user to *drag* the mouse while picking which is as quite unusual use for picking

In rare cases volume data that results in a DVR with opacity varying with a high frequency along the ray can influence the picking. In such cases the detection using the second derivative can be the problem because γ^{acc} reacts strongly to high-frequency changes. This problem did appear only rarely in the tested real-world data sets and disturbed the picking only marginally but will be addressed in future work with a smoothing kernel nevertheless.

However, as demonstrated in the video, these limitations in general do not impede the picking.

7 Conclusion and Future Work

We have presented a method to pick positions observed in volumetric renderings of three-dimensional data in an intuitive manner. In contrast to previous methods, the described approach is perception-oriented and thus is applicable for any type of volume rendered data. It only uses the transfer function of the volume rendering together with the data itself (of course) to determine the opacity and thus the visible features along the viewing direction. Observable features are characterized by large jumps in the accumulated opacity; the picked feature corresponds to the largest jump of the accumulated opacity. We emphasized the fact that no metadata is needed by demonstrating the method with data from a flow simulation where no metadata is available. The usefulness of the proposed technique for medical data has been shown by its application to an abdominal MRI scan. The application to flow and other data shows that the method is useful far beyond the medical scope.

As mentioned before, the perception-oriented picking has been developed for volume rendering without local illumination, research into picking in illuminated direct volume rendering is one of the next steps. Furthermore, we are already working on incorporating information from rays in the vicinity of the ray through the picked position. This will also contribute to the technique dealing with illuminated rendering. We will have to employ more computer vision techniques [Mar82] to deal with the information from the additional rays.

Acknowledgments

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) of the VIGOR++ Project under grant agreement nr. 270379. The information presented is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability. The opinions expressed in the document are of the authors only and in no way reflect the European Commission's opinions. Except where indicated, all images were produced with Amira [SWHCH05]. The authors would like to thank Markus Rütten from German Aerospace Center (DLR) in Göttingen for providing help and code for the flow simulation. Special thanks got to David Foerster for his help with the implementation.

References

[AGI*08] AGUS M., GOBBETTI E., IGLESIAS GUITIÁN J., MARTON F., PINTORE G.: GPU accelerated direct volume rendering on an interactive light field display. *Computer Graphics Forum (EUROGRAPHICS 2008) 27*, 3 (2008), 231–240.

[Avi] Avizo - the 3D analysis software for scientific and industrial data. http://www.vsg3d.com/avizo/overview, last visited 2011-09-21.

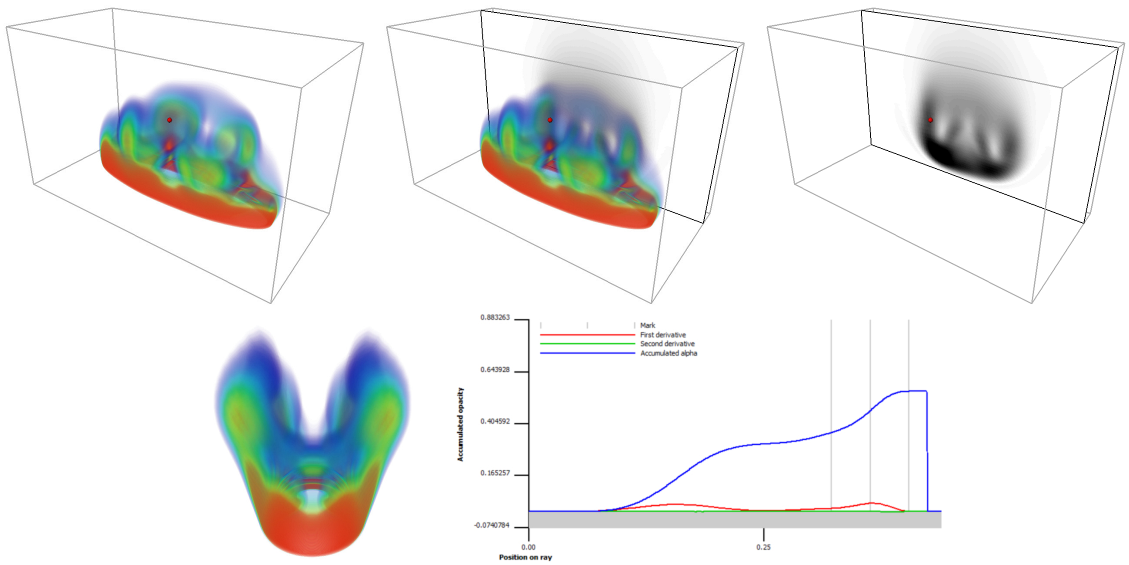

Figure 7: Picking in DVR of vorticity field of flow around an ellipsoid. The images in the upper row show the picking process (picked position as red dot). The lower left image shows the volume rendered data from a different perspective. The lower right shows the accumulated opacity and its derivatives along the ray. Curves of derivatives are scaled by a factor of ten, but changes in second derivative are still hardly visible.

[BH92] BIDGOOD W. D., HORII S. C.: Introduction to the ACR-NEMA DICOM standard. *Radiographics a review publication of the Radiological Society of North America Inc 12*, 2 (1992), 345–355.

[BKKG08] BRUCKNER S., KOHLMANN P., KANITSAR A., GRÖLLER M. E.: Integrating volume visualization techniques into medical applications. In *Proceedings of 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro* (May 2008), pp. 820–823.

[BS05] BORDOLOI U. D., SHEN H.-W.: View selection for volume rendering. In *IEEE Visualization 2005* (2005), pp. 487 – 494.

[BŠG*09] BRUCKNER S., ŠOLTÉSZOVÁ V., GRÖLLER E., HLADŮVKA J., BÜHLER K., YU J., DICKSON B.: Braingazer - visual queries for neurobiology research. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (Nov. 2009), 1497–1504.

[CM09] CORREA C. D., MA K.-L.: Visibility-driven transfer functions. In *Proceedings of the 2009 IEEE Pacific Visualization Symposium* (Washington, DC, USA, 2009), PACIFICVIS '09, IEEE Computer Society, pp. 177–184.

[CSA00] CARR H., SNOEYINK J., AXEN U.: Computing contour trees in all dimensions. In *SODA '00: Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms* (Philadelphia, PA, USA, 2000), Society for Industrial and Applied Mathematics, pp. 918–926.

[CWM*09] CHAN M.-Y., WU Y., MAK W.-H., CHEN W., QU H.: Perception-based transparency optimization for direct volume rendering. *IEEE Transactions on*

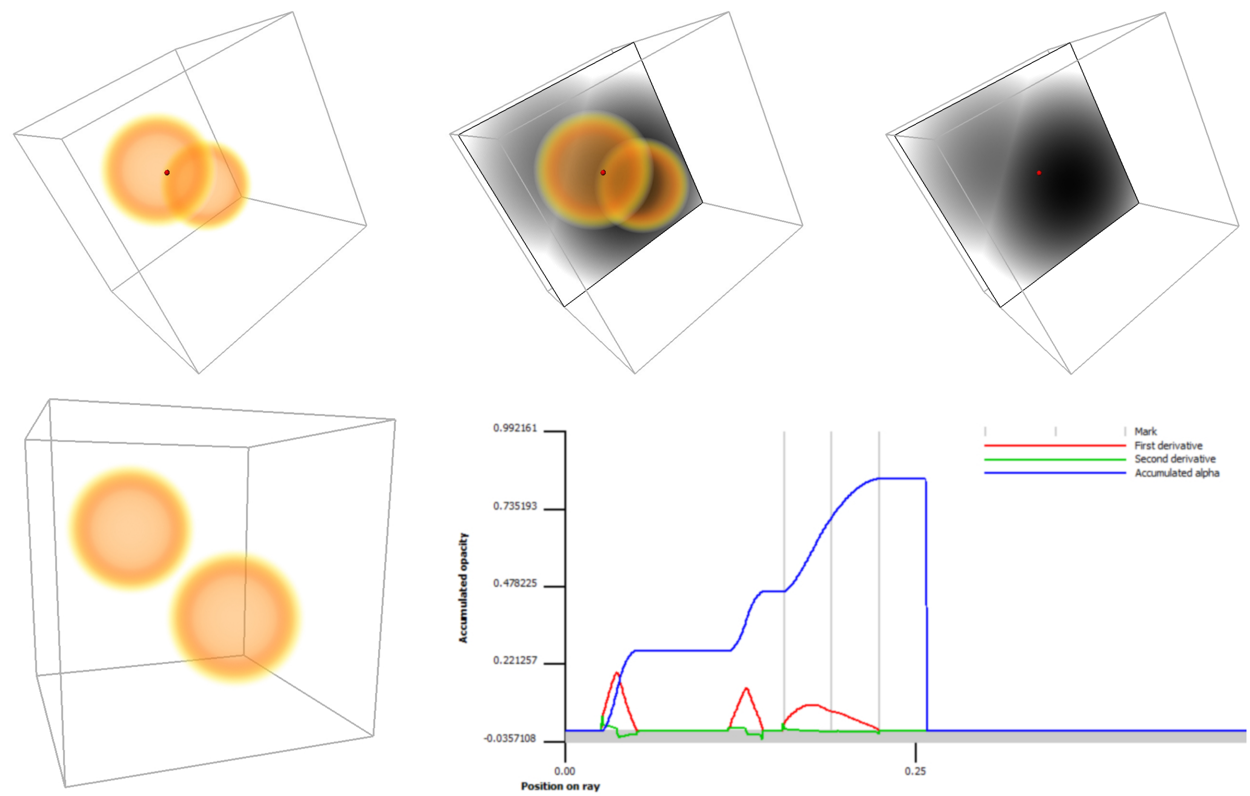

Figure 8: Picking in synthetic data shows that a stronger region in background can be picked well. The images in the upper row show the picking process (picked position as red dot). The lower left image shows the volume rendered data from a different perspective. The lower right shows the accumulated opacity and its derivatives along the ray. Curves of derivatives are scaled by a factor of ten.

Visualization and Computer Graphics 15 (November 2009), 1283–1290.

[EHK*06] ENGEL K., HADWIGER M., KNISS J., REZK-SALAMA C., WEISKOPF D.: *Real-Time Volume Graphics*. AK Peters, 2006.

[GPZT98] GOBBETTI E., PILI P., ZORCOLO A., TUVERI M.: Interactive virtual angioscopy. In *Proceedings of the conference on Visualization '98* (Los Alamitos, CA, USA, 1998), VIS '98, IEEE Computer Society Press, pp. 435–438.

[HE11] HEALEY C. G., ENNS J. T.: Attention and visual memory in visualization and computer graphics. *IEEE Transactions on Visualization and Computer Graphics 99*, PrePrints (2011).

[KBKG07] KOHLMANN P., BRUCKNER S., KANITSAR A., GRÖLLER M. E.: LiveSync: Deformed viewing spheres for knowledge-based navigation. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (Oct. 2007), 1544–1551.

[KBKG08] KOHLMANN P., BRUCKNER S., KANITSAR A., GRÖLLER M. E.: LiveSync++: Enhancements of an interaction metaphor. In *Proceedings of Graphics Interface 2008* (May 2008), pp. 81–88.

[KBKG09] KOHLMANN P., BRUCKNER S., KANITSAR A., GRÖLLER M. E.: Contextual picking of volumetric structures. In *Proceedings of the IEEE Pacific Visualization Symposium 2009* (5 2009), Peter Eades Thomas Ertl H.-W. S., (Ed.), pp. 185–192.

[KW03] KRÜGER J., WESTERMANN R.: Acceleration techniques for GPU-based volume rendering. In *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)* (Washington, DC, USA, 2003), VIS '03, IEEE Computer Society, pp. 287–292.

[LHH*08] LEE K., HONG H., HAHN S., KIM B., KIM K., KIM Y.: Summation or axial slab average intensity projection of abdominal thin-section ct datasets: Can they substitute for the primary reconstruction from raw projection data? *Journal of Digital Imaging 21* (2008), 422–432. 10.1007/s10278-007-9067-y.

[Mar82] MARR D.: *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. W. H. Freeman, 1982.

[Max95] MAX N.: Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics 1* (1995), 99–108.

[ME04] MORA B., EVERT D. S.: Instant Volumetric Understanding with Order-Independent Volume Rendering. *Computer Graphics Forum (EUROGRAPHICS 2004) 23*, 3 (2004), 489–497.

[MMG07] MALIK M. M., MÖLLER T., GRÖLLER E.: Feature peeling. In *Proceedings of Graphics Interface 2007* (Favoritenstrasse 9-11 / E186, A-1040 Wien, Austria, May 2007), A K Peters Ltd, pp. 273–280.

[MSRMH09] MEYER-SPRADOW J., ROPINSKI T., MENSMANN J., HINRICHS K. H.: Voreen: A rapid-prototyping environment for ray-casting-based volume visualizations. *IEEE Computer Graphics and Applications (Applications Department) 29*, 6 (Nov./Dec. 2009), 6–13.

[MTRP10] MÜHLER K., TIETJEN C., RITTER F., PREIM B.: The medical exploration toolkit - an efficient support for visual computing in surgical planning and training. *IEEE Transactions on Visualization and Computer Graphics 16*, 1 (2010), 133–146.

[MVL] MeVisLab - development environment for medical image processing and visualization. http://www.mevislab.de, last visited 2011-09-21.

[MWCQ11] MAK W.-H., WU Y., CHAN M.-Y., QU H.: Visibility-aware direct volume rendering. *J. Comput. Sci. Technol. 26* (March 2011), 217–228.

[NCP] VTK/New CellPicker - for vtkVolume objects. http://www.vtk.org/Wiki/VTK/New_CellPicker, last visited 2011-09-21.

[PB07] PREIM B., BARTZ D.: *Visualization in Medicine*. Morgan Kaufman, 2007.

[PRL*10] PENG H., RUAN Z., LONG F., MYERS J. H., MYERS E. W.: V3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets. *Nat Biotech 28*, 4 (2010), 348–353.

[PSL*98] PARKER S., SHIRLEY P., LIVNAT Y., HANSEN C., SLOAN P.-P.: Interactive ray tracing for isosurface rendering. In *Proceedings of the conference on Visualization '98* (Los Alamitos, CA, USA, 1998), VIS '98, IEEE Computer Society Press, pp. 233–238.

[RSK06] REZK-SALAMA C., KOLB A.: Opacity peeling for direct volume rendering. *Computer Graphics Forum 25*, 3 (2006), 597–606.

[Sab88] SABELLA P.: A rendering algorithm for visualizing 3D scalar fields. *SIGGRAPH Comput. Graph. 22* (June 1988), 51–58.

[SML98] SCHROEDER W., MARTIN K., LORENSEN B.: *The Visualization Toolkit*, 2nd ed. Prentice Hall, 1998.

[Squ08] SQUILLACOTE A.: *The ParaView Guide*, 3rd ed. Kitware, Inc., 2008.

[SRP91] SIEBERT J., ROSENBAUM T., PERNICONE J.: Automated segmentation and presentation algorithms for 3D MR angiography. *Book of Abstracts, SMRM* (1991).

[SWHCH05] STALLING D., WESTERHOFF M., HANS-CHRISTIAN-HEGE: Amira: A highly interactive system for visual data analysis. In *The Visualization Handbook*, Hansen C. D., Johnson C. R., (Eds.). Elsevier, 2005, pp. 749–767.

[TD97] TOENNIES K. D., DERZ C.: Volume rendering for interactive 3-d segmentation. In *Proceedings of the SPIE (Medical Imaging 1997)* (1997), vol. 3031, pp. 602–609.

[TM04] TORY M., MÖLLER T.: Human factors in visualization research. *IEEE Transactions on Visualization and Computer Graphics 10* (January 2004), 72–84.

[Vli08] VLIETINCK J.: Picking on fused 3D volume rendered images and updating corresponding views according to a picking action, 2008. Patent No EP1923840A2.

[WMLK89] WALLIS J., MILLER T., LERNER C., KLEERUP E.: Three-dimensional display in nuclear medicine. *IEEE Transactions on Medical Imaging 8*, 4 (Dec 1989), 297 – 230.

[WZC*11] WANG Y., ZHANG J., CHEN W., ZHANG H., CHI X.: Efficient opacity specification based on feature visibilities in direct volume rendering. *Computer Graphics Forum (Pacific Graphics 2011) 30*, 7 (2011), 2117–2126.