Dynamic Focus+Context for Volume Rendering

P. Sikachev[1], P. Rautek[1], S. Bruckner[1] and M. E. Gröller[1]

[1]Institut für Computergraphik und Algorithmen, TU Vienna, Austria

Abstract

Interactive visualization is widely used in many applications for efficient representation of complex data. Many techniques make use of the focus+context approach in a static manner. These techniques do not fully make use of the interaction semantics. In this paper we present a dynamic focus+context approach that highlights salient features during user interaction. We explore rotation, panning, and zooming interaction semantics and propose several methods of changing visual representations, based on a suggested engagement-estimation method. We use DVR-MIP interpolation and a radial opacity-change approach, exploring rotation, panning, and zooming semantics. Our approach adds short animations during user interaction that help to explore the data efficiently and aid the user in the detection of unknown features.

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.3]: Display algorithms—; Computer Graphics [I.3.3]: Viewing algorithms—

1. Introduction

In typical visualization scenarios, users focus on particular characteristics of the data. These features are often shown together with contextual features in the proximity. Such visualizations employ *focus+context* techniques, and the main goal is to emphasize *focus* information, while *context* should provide a reference for it.

Focus+context methods are widely used in visualization systems. Although they are very well suited for many situations, the full human visual bandwidth could be exploited even further. Many approaches address this issue using animation. Animation is very useful to highlight the focus of visualization and also to visualize data quantities such as uncertainty [LLPY07]. However, an excessively animated scene may distract the user's attention and lead to an overall lower effectiveness of the visualization. This observation leads us to the design of our novel dynamic focus+context technique that carefully uses animation only during and shortly after interaction. Our assumption is that the user needs guidance most during user interaction. This is the time the user is either searching new features in the data or looking for alternative perspectives on the data. Figure 1 shows a comparison of the discussed approaches. The conventional focus+context model renders the focus with a constant parameter. The animated focus+context model animates the visualization, effectively changing the focus parameter over time

(e. g., the work of Woodring and Shen [WS07]). A typical way to animate the visualization is to change the opacity, or color of the focus as well as the context. Other approaches have shown that also animation of the objects' transformation is useful [HR07]. Once the animation is stopped, the animation model offers the option to either loop or run in reverse. This choice along with specifying other parameters such as the speed of the animation and the duration between consecutive runs of the animation have to be made either when designing the visualization techinque or by the user. In contrast our novel dynamic focus+context model offers a simple method to incorporate animations in a user-guided, non obtrusive way. Animations are only run at the beginning and the end of user interactions. With this model it is easy for the user to frequently switch between the animation and the visualization mode.

We propose a technique that tries to anticipate the region of interest, i.e., the *focus* object or data. Our technique animates the rendering style (which is denoted by a *style function*) of each voxel in correspondence with the chosen user interaction. Sometimes 2D/3D techniques change the rendering style of objects during interaction. For instance, in 3D editors during rotation the objects are shown in wireframe mode. This pursuits two particular goals. First, on slow computers higher frame rates are achieved. Second, the user can
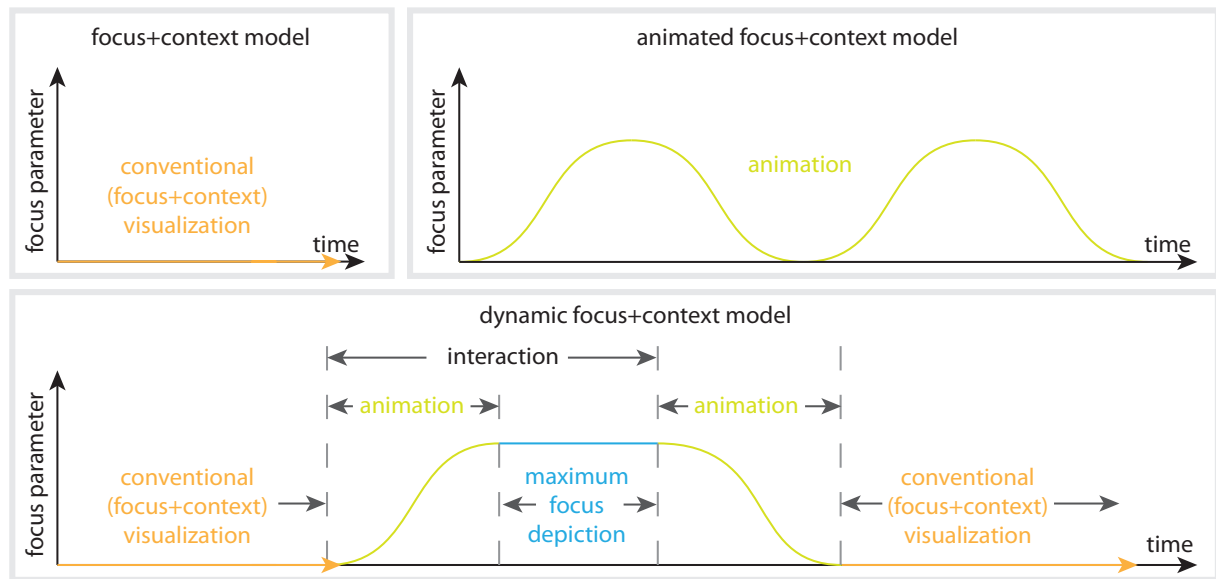
Figure 1: *Focus+context models. The traditional focus+context model is shown at the top left, the technique Woodring and Shen [WS07] is shown at the top right, and the proposed model is shown at the bottom. Note, that unlike previous models, our model is highly dependent on user interaction.*

better explore the scene and abstract from unnecessary details.

Besides a style change, in 2D applications like Google maps, during scaling the focus is modified according to the position of the mouse cursor. When the user is zooming, the mouse cursor is assumed to be close to the region, that he wants to investigate. Therefore, during the scaling operation, the *focus* point starts to shift towards the center of the viewport. We also make use of this type of *focus* change during user interaction.

Figure 2 shows an example to illustrate dynamic focus+context approach. The process of zooming on a particular feature is shown. From left to right, the user starts to zoom on the cheek of the face, interior details are revealed. After the user stops the interaction, a short animation reverts to the initial visualization mode, which is why the interior of the object is not shown anymore.

The purpose of this work is to develop a general concept of a system that changes the visualization style of the object to help users explore additional features, during interaction. In this paper we investigate three basic user interactions, i.e., rotation, zooming and panning. We describe several dynamic focus+context visualization techniques.

The remainder of the paper is organized as follows. In Section 2 we give an overview of related techniques. In Section 3 we describe the basics of our approach. In Sections 4-7 we describe each of the modules of our approach in detail.

We discuss our results in Section 8 and conclude in Section 9.

2. Related Work

Several approaches introduce a combination of different rendering techniques to achieve focus+context visualizations. For instance, mixing pen-and-ink illustrations and direct volume rendering (DVR) is shown in the work of Treavett and Chen [TC00]. A method described by Interrante at al. [IFP96] involves the representation of the outer context with strokes which do not take up much image-space. Also some efforts were made in order to generalize combinations of various visual representations [HMBG01, LM02].

In information visualization one approach is to use a semantic depth-of-field [KMH01]. In this technique the objects of interest are shown sharply while other objects stay out of focus and appear blurry. *Smart visibility* techniques use visual distortion in order to better represent the *focus* object [CSC*96, VKG04].

Woodring and Shen [WS07] demonstrate the enhancement of certain features of objects using animation. They effectively show how motion draws the attention of the user towards preselected areas of interest. Their approach is used to highlight regions that correspond to selected data ranges. We build upon their findings that animation helps to direct the user's attention. A continuously running animation, however, might distract the user and also might prevent a closer inspection of the focus as well as the context. Therefore we
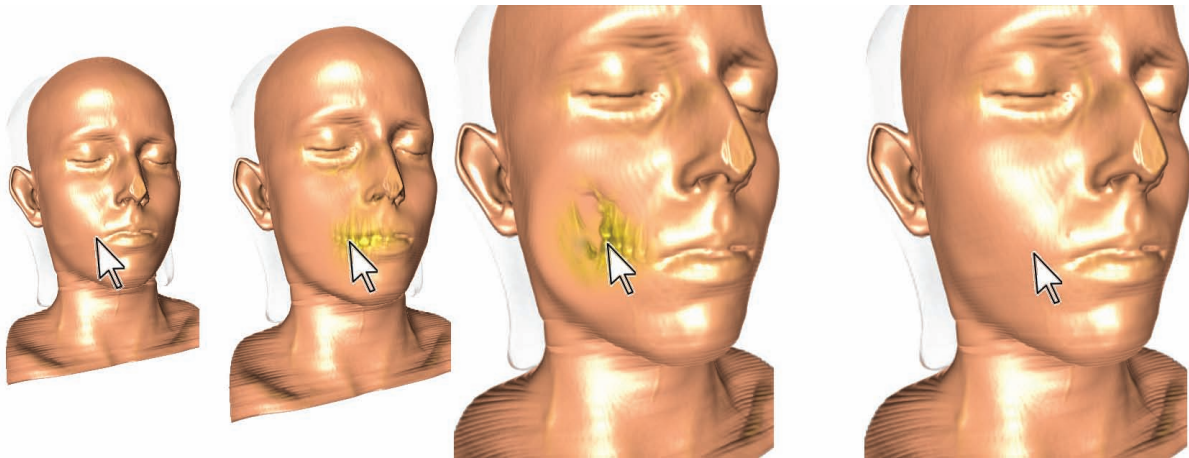
Figure 2: *Zooming on the cheek. A series of screenshots showing the dynamic focus+context approach during user interaction. The opacity in the focus region is animated to unveil additional features during interaction. The mouse cursor is shown for reference purposes.*

use animation only during interaction. Our approach also does not necessarily depend on a pre-selected region, but anticipates the focus regions based on several assumptions.

Bruckner and Gröller [BG07] propose using halos in order to make the structure of interest stand out. In their method, regions near contours of the objects are either brightened or darkened in order to highlight the object. Bruckner et al [BGKG06] investigate how to show both exterior and interior features while preserving the context.

Joshi and Rheingans [JR05] proposed a technique for a temporally exploded view illustration. They use speedlines and strobe silhouettes to depict motion and previous position respectively. In contrast to their approach, we change the visualization style in order to reveal additional details, and do not focus on motion illustration.

Krüger et al. [KSW06] present a highly interactive focus+context visualization system that enables the rendering of additional features in a focus region. In their approach a cursor is used to indicate the position of interest. Chen et al. [CTM08] describe a technique for interactively magnifying regions of interest in the data, that allows to change the focus of the visualization and to explore large data sets. With our approach we want to enhance existing interaction techniques (like zooming, panning, and rotation) without the need for additional user interaction. Further, we want to switch between the focus and a more contextual rendering in an unobtrusive way. Along these lines Kohlmann et al. [KBKG07] proposed an approach for the interactive synchronization of two visualizations for the special case of medical visualizations in clinical practice. Although the idea of using an additional interaction is similar, the focus of our work is the enhancement of interaction techniques in a more general way. Viola et al. [VFSG06] describe a method for

the automatic generation of good viewpoints on pre-defined focus regions. Animations are presented to visit the focus regions of interest. We want to achieve similar goals as shown in their work, however only for the purpose of enhancing the user interaction without the knowledge of pre-defined regions. StyleCam presented by Burtnyk et al. [BKF*02] is an approach that allows the authoring of camera movements, and user interactions with cameras. A constraint movement of cameras on camera surfaces is combined with animations for continuous movements between surfaces. The authors describe a state diagram for the user interaction similar to our approach. However, we do not focus on authoring of constrained camera interactions, but on animations during user interaction.

In information visualization many techniques were proposed to interactively change the focus+context of visualization. Carpendale et al. [CSC*96] show a technique for distorting the 3D layout of a set of objects to reveal the focus. Van Wijk and Nuij [vWN04] present a general model for the smooth navigation in large 2D scenes. They also propose methods for non-linear zooming and rotation. Our approach extends some of the mentioned techniques to 3D, for the special case of volume visualization.

3. Overview of the Framework

The outline of our framework is shown in Figure 3. The interaction module encapsulates the typical user interaction activities, such as rotation, panning and zooming. This module can also involve semantic-dependent navigation, e. g. adjusting the point the virtual camera is focusing at during zooming, as described in Section 4. The viewing transformation matrix is dependent not only on the user input, but it is also
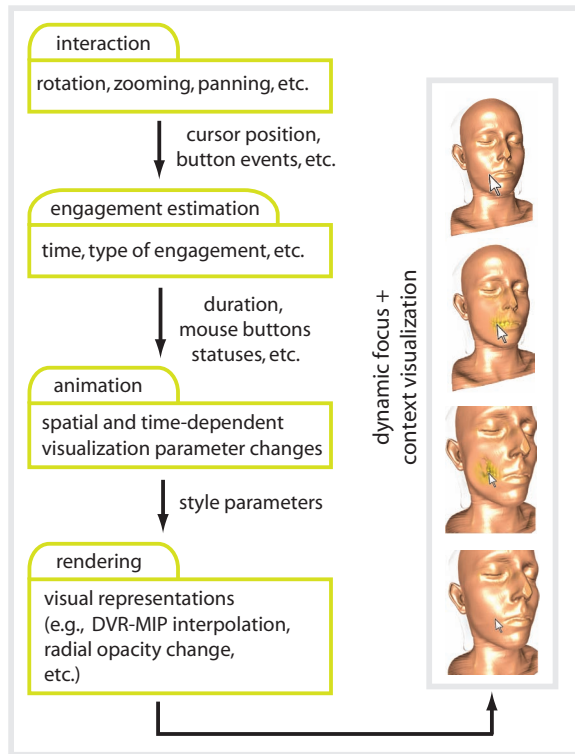
Figure 3: *Outline of the presented dynamic focus+context framework: the interpretation of interaction guides the enhancement of the visualization.*

modified in a way to reveal additional information. For example, during zooming it can be adjusted, to move a region of interest closer to the center of the viewpoint.

The engagement estimation module processes the interaction information, evaluating the type of engagement and its parameters. The number and semantics of the output parameters depend on the interaction type. The engagement estimation module can also incorporate further statistics, like the duration of interaction with a particular region of an object from viewpoint (that occurs in the process of zooming), or other interaction patterns, as rotation around the axis. It also evaluates the speed of interaction and can make predictions about consecutive viewpoints.

The animation module changes the style of each voxel over time, by defining style parameters. This function returns a structure of flags and parameters for the focus+context visualizations. The flags define extra features as cut-aways, while the factors (e. g. the style function value) define contributions of different rendering styles to the final result. The style function is defined for the arbitrary point of the scene, visible to the user.

The rendering module implements the visual representa-

tions and determines the final appearance of the rendered data. It uses style parameters, obtained from the previous stage, as input to the volume rendering techniques. Apart from rendering itself, it can also estimate visibility from different viewpoints, predicted by the engagement estimation module. This information could be used either to modify the transfer function or to add auxiliary information about what the user can see from another viewpoint. Each module is discussed in more detail in the consecutive sections.

4. Interaction

In the current implementation, the interaction module processes information about three types of interactions: panning, zooming and rotation. It passes mouse position and button events to the engagement estimation module. The rotation and panning interactions are based on a trackball metaphor [CMS88].

For zooming interaction we use a model for a shift of the center of the object. We assume that during a zooming operation the user is most interested in the parts of the object near the initial button click. Equation 1 denotes such a transform, where MV is the original model-view matrix and MV' is the modified model-view matrix, R denotes the magnification factor, $\vec{P}_{interaction}$ and $\vec{P}_{center}$ denote the position of the initial user interaction and center of the rendering area respectively. $Translate$ is a transformation matrix that shifts an object by a specified vector.

$$MV' = Translate(-R \cdot (\vec{P}_{interaction} - \vec{P}_{center})) \cdot MV \quad (1)$$

This transformation allows to emphasize user's attention on the *focus* parts of the object. For panning and rotation we do not use additional semantic-dependent navigation, although this may be easily added if needed.

5. Engagement Estimation

We use three engagement-estimation modules for the three interaction types involved. For rotation, panning and zooming the time of engagement is calculated using a timer. When the user finishes the interaction, the timer is zeroed, but still used to evaluate how long the user interacted the last time.

For the zooming interaction, this module also outputs coordinates of the user-interaction point. The user-interaction point is actually the position of initial interaction on the object.

6. Animation

In our system the animation module outputs a floating point value in the interval [0, 1]. For example, it can indicate a degree of user's interaction with 0 corresponding to the situation when a user does not interact with the object and 1

- when a user interacts for a very long time. In the case of rotation, this value is the same for the whole object. As an estimation of user interaction, we use an exponential function, shown in Equation 2. $S(t_{current})$ is the current style function value, $S(t_{start})$ is the style function value before beginning or ending of the interaction. $S(t_{dest})$ is 0 if a user stopped interaction and 1 if a user started interacting.

$$S(t_{current}) = S(t_{start}) \cdot e^{-(t_{current} - t_{start})} + \qquad (2)$$
$$S(t_{dest}) \cdot (1 - e^{-(t_{current} - t_{start})})$$

The purpose of that this kind style function is to provide a time-varying coefficient for modifying the rendering style of the object or its *focus* part. As can be observed from Equation 2, when the user starts the interaction, the style function smoothly changes towards $S(t_{dest})$. When the user finishes the interaction, the style function smoothly changes back to its original value. That gives an opportunity to control the visualization parameters according to the user's involvement into interaction.

In case of zooming a spatially-dependent factor is also transferred to the rendering module. The style parameters are adjusted, so that the closer a projection of a voxel is to the interaction point, the higher the style-function value is.

7. Visual Representations

In our work we present three visual representations: DVR-MIP interpolation, radial opacity change and focus translation. These techniques are described in detail below.

7.1. DVR-MIP Interpolation

A transition between direct volume rendering (DVR) and maximum intensity projection (MIP) was suggested by Bruckner and Gröller [BG09]. A parameter C, that changes from -1 to 1 is used, where -1 corresponds to DVR, 0 corresponds to maximum intensity difference accumulation (MIDA) and 1 corresponds to DVR. The style function S, proposed above, is mapped to this domain, using Equation 3.

$$S(t) = \frac{C(t) + 1}{2} \qquad (3)$$

The usage of this function allows to reveal inner details (using the MIP renderer) during interaction. A typical scenario of using this technique is shown in Figure 4.

Figure 5 shows DVR-MIP interpolation for a human body. The user starts to rotate the object, trying to find vascular structure in the leg. Due to the continuous interaction, the visual representation changes, showing the focus information to the user.
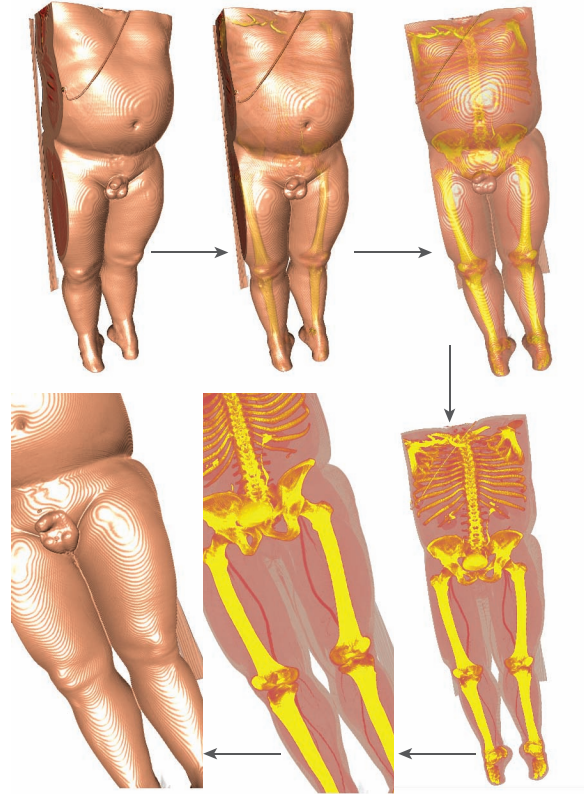
Figure 5: *DVR-MIP interpolation for a full body scan. Notice how our approach helps to reveal the vascular structure that is not visible before the interaction.*

7.2. Radial Opacity Change

The radial opacity change is based on the angular velocity. This model is inspired by the metaphor of a viscous quickly rotating around its center. The speed of particles at the border is highest, such that they are candidates to disappear from the object.

The style function is used here as an exponent, which is shown in Equation 4. Here α stands for the opacity (alpha-value), S stands for a style function, obtained from the previous step and $\rho(\vec{P}_{voxel}, \vec{P}_{center})$ is the distance from the voxel to the center of the object.

$$\alpha_{voxel} = e^{-S(t) \cdot \rho(\vec{P}_{voxel}, \vec{P}_{center})} \qquad (4)$$

This method helps to reveal inner parts of the object during rotation. The further a voxel is from the center of the object, the more transparent it becomes, due to $\rho(\vec{P}_{voxel}, \vec{P}_{center})$ coefficient.

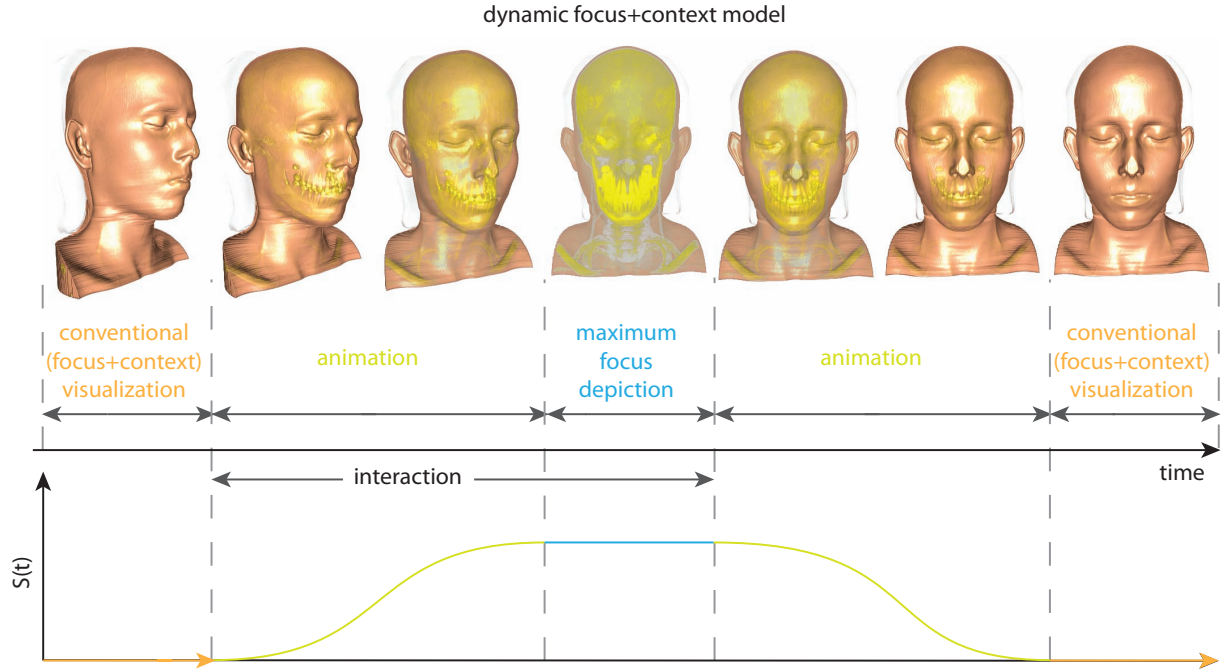For the zooming interaction the same approach can be ex-

dynamic focus+context model

Figure 4: *Dynamic focus+context via DVR-MIP interpolation. A typical scenario is shown, where the user starts interaction, interacts for some time and then stops the interaction.*

ploited as well. Equation 5 shows the result of combining Equation 4 and a spatially-dependent factor. Here $\vec{P}_{focus}$ denotes the initial point of interaction on the object.

$$\alpha_{voxel} = e^{-S(t) \cdot max(0, \rho(\vec{P}_{voxel}, \vec{P}_{center}) - \rho(\vec{P}_{voxel}, \vec{P}_{focus}))} \quad (5)$$

The $\rho(\vec{P}_{voxel}, \vec{P}_{focus})$ term makes the inner parts of the object better visible close to the *focus* point, leaving the *context* around untouched.

Figure 2 shows radial opacity change technique, applied to the zooming interaction. The user is zooming on the cheek and the inner details are revealed near the initial point of interaction, while this point is shifted towards the center of the viewport.

7.3. Focus Translation

With the focus translation we propose a tool for feature investigation while the user is panning the object. It is inspired by an analogy to the effect of inertia on bodies: if a non-uniform body is moved its more dense parts will accelerate and decelerate slower than less dense parts.

We implement this technique by modifying the coordinates of the volume samples for the denser parts of the object which are obtained using a threshold value. Let $\vec{P}$ be the current object position, $\vec{P}_{SI}$ the object position at the start of

interaction, then $\vec{P}_{dense}$ is the output position for the denser points, and $S(t)$ is the style function, with $S(t_{start}) = 0$ and $S(t_{dest}) = 1$. The longer the user is panning the object, the further dense regions are displaced.

$$\vec{P}_{dense} = \vec{P} + (\vec{P} - \vec{P}_{SI}) \cdot S(t) \quad (6)$$

When the user stops the interaction, the denser region moves back to its initial position within the object. Equation 7 shows how this process is implemented. All parameters are the same as in Equation 6 except $\vec{P}_{SI*}$ denotes the position of the denser parts of the object at the moment when the interaction stopped and $S^*(t)$ is the style function with $S^*(t_{dest}) = 0$, and t_{start} equal to the $t_{current}$ and $S^*(t_{start})$ equal to the $S(t)$ values at the moment when interaction finished.

$$\vec{P}_{dense} = \vec{P} + (\vec{P} - \vec{P}_{SI*}) \cdot S^*(t) \quad (7)$$

Figure 6 illustrates the process of panning. The user is panning the object to the left and the denser parts of the object are standing out. When the interaction is finished, the object returns to the proper position.

8. Discussion

We presented a novel model for interaction with volumetric data. Our approach proposes to more efficiently make use of the human visual bandwidth, by using animations during
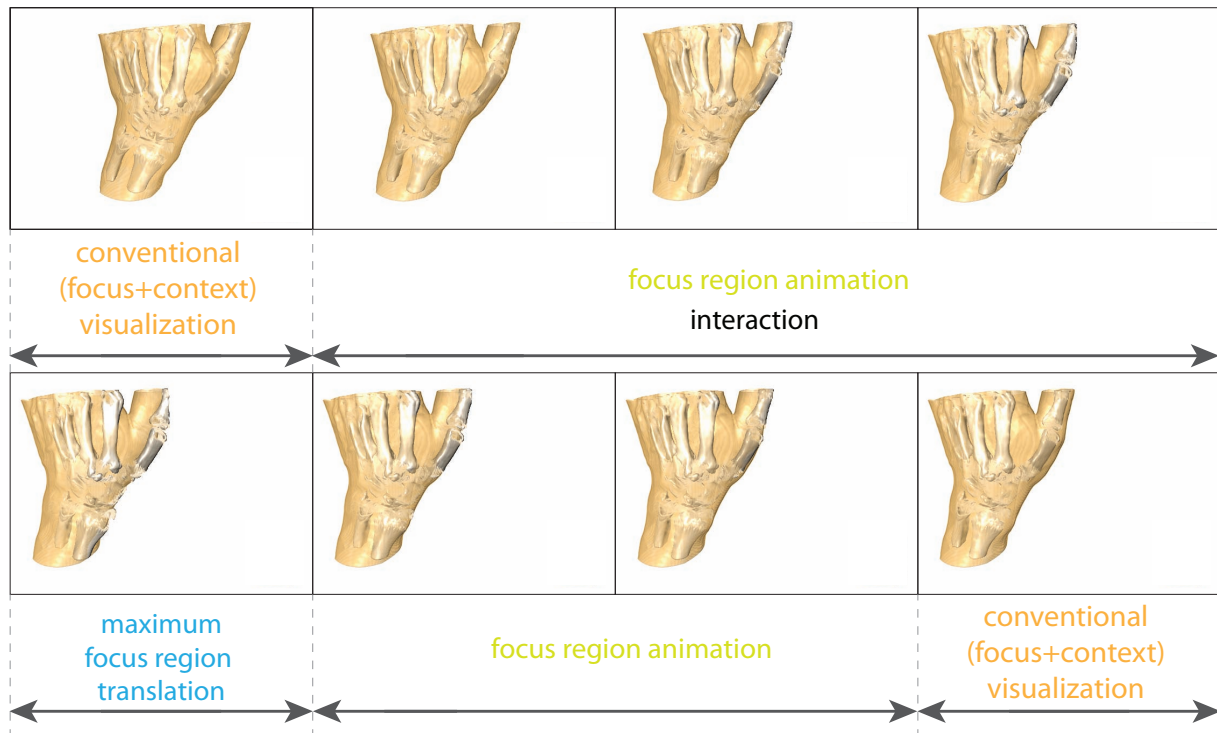
Figure 6: *Inertion-based feature investigation in a human hand. Notice how the bones begin protrude while the user shifts the object along the horizontal axis.*

user interactions. We assume that phases of user interactions are a good target for enhancing visualizations for three reasons:

- Phases of user interacting (hypothesizing a data exploration scenario) are phases of search for additional features.
- Interactions are typically short and often repetitive tasks. With short animated changes of the visualization style, the user's attention is not distracted and the visualization quickly returns to the original style.
- User interactions can easily be used to trigger the animations.

In this paper we did not focus on the evaluation of the proposed concept. As stated by Cockburn et al. [CKB08], "[...] *the current state of research fails to provide clear guidelines* [..., since] *the efficiency of different techniques is dependent on many factors, particularly the nature of the users' task*". The goal of our work was not to present a novel technique that is proven to be more efficient for a very specific task, but to present an alternative model that can greatly enhance the user's experience. However, to get an initial indication on the impact of the presented techniques, we performed a small-scale pilot study which included two types of formal tasks and a short survey. In a constrained setup we measured user performance for two

simple visual tasks using our approach and compared it to manual adjustment of rendering parameters using a slider widget. While our method could not achieve significant improvements in terms of performance or error rate, several users commented that they felt more confident when interacting using the proposed concept and that they had to spend less time on switching between interaction modalities. In particular in the context of 3D scientific visualization applications, which typically feature many complex user interface elements, we therefore believe that the integration of the proposed approach could help to improve the user experience. A detailed description of the user study and its results can be found at: http://www.cg.tuwien.ac.at/research/publications/2010/sikachev-2010-DFC/

9. Conclusion and Future Work

In this paper, we presented a model for dynamic focus+context in volume visualization. We discussed several visual representations (DVR-MIP interpolation, radial opacity change) and interaction types (rotation, zooming, panning) based on our concept. In the future we will investigate additional interaction types as well, e.g., rotation around an axis. Besides radial opacity change and DVR-MIP interpolation, other methods, as transfer function adjustment, will be explored for better revealing focus. Input semantics of other

input devices are another direction of investigation. Further, we research a viewpoint prediction model. This model estimates the next viewpoint and parts of the object, that are better visible from it. During rotation along axis, the parts of the object that would become visible from other viewpoints can be shown to guide this rotation.

10. Acknowledgements

References

[BG07] BRUCKNER S., GRÖLLER E.: Enhancing depth-perception with flexible volumetric halos. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (2007), 1344–1351.

[BG09] BRUCKNER S., GRÖLLER M. E.: Instant volume visualization using maximum intensity difference accumulation. *Comput. Graph. Forum 28*, 3 (2009), 775–782.

[BGKG06] BRUCKNER S., GRIMM S., KANITSAR A., GRÖLLER M. E.: Illustrative context-preserving exploration of volume data. *IEEE Transactions on Visualization and Computer Graphics 12*, 6 (2006), 1559–1569.

[BKF*02] BURTNYK N., KHAN A., FITZMAURICE G., BALAKRISHNAN R., KURTENBACH G.: StyleCam: interactive stylized 3d navigation using integrated spatial & temporal controls. In *UIST '02: Proceedings of the 15th annual ACM symposium on User interface software and technology* (New York, NY, USA, 2002), ACM, pp. 101–110.

[CKB08] COCKBURN A., KARLSON A., BEDERSON B. B.: A review of overview+detail, zooming, and focus+context interfaces. *ACM Comput. Surv. 41*, 1 (2008), 1–31.

[CMS88] CHEN M., MOUNTFORD S. J., SELLEN A.: A study in interactive 3-d rotation using 2-d control devices. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1988), ACM, pp. 121–129.

[CSC*96] CARPENDALE M., SHEELAGH T., CARPENDALE T., COWPERTHWAITE D. J., FRACCHIA F. D.: Distortion viewing techniques for 3-dimensional data. In *INFOVIS '96: Proceedings of the 1996 IEEE Symposium on Information Visualization (INFOVIS '96)* (Washington, DC, USA, 1996), IEEE Computer Society, p. 46.

[CTM08] CHEN C.-K., THOMASON R., MA K.-L.: Intelligent focus+context volume visualization. vol. 1, IEEE Computer Society, pp. 368–374.

[HMBG01] HAUSER H., MROZ L., BISCHI G. I., GRÖLLER E.: Two-level volume rendering. *IEEE Trans. Vis. Comput. Graph. 7*, 3 (2001), 242–252.

[HR07] HEER J., ROBERTSON G.: Animated transitions in statistical data graphics. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (2007), 1240–1247.

[IFP96] INTERRANTE V., FUCHS H., PIZER S. M.: Illustrating transparent surfaces with curvature-directed strokes. In *VIS '96: Proceedings of the conference on Visualization '96* (1996), pp. 211–218.

[JR05] JOSHI A., RHEINGANS P.: Illustration-inspired techniques for visualizing time-varying data. In *VIS '05: Proceedings of the conference on Visualization '05* (2005).

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

[KMH01] KOSARA R., MIKSCH S., HAUSER H.: Semantic depth of field. In *INFOVIS '01: Proceedings of the 2001 IEEE Symposium on Information Visualization (INFOVIS '01)* (2001), pp. 97–104.

[KSW06] KRÜGER J., SCHNEIDER J., WESTERMANN R.: Clearview: An interactive context preserving hotspot visualization technique. *IEEE Transactions on Visualization and Computer Graphics 12* (2006), 941–948.

[LLPY07] LUNDSTRÖM C., LJUNG P., PERSSON A., YNNERMAN A.: Uncertainty visualization in medical volume rendering using probabilistic animation. *IEEE Transactions on Visualization and Computer Graphics 13* (2007), 1648–1655.

[LM02] LUM E. B., MA K.-L.: Hardware-accelerated parallel non-photorealistic volume rendering. In *NPAR* (2002).

[TC00] TREAVETT S. M. F., CHEN M.: Pen-and-ink rendering in volume visualisation. In *VIS '00: Proceedings of the conference on Visualization '00* (2000), pp. 203–210.

[VFSG06] VIOLA I., FEIXAS M., SBERT M., GRÖLLER M. E.: Importance-driven focus of attention. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006), 933–940.

[VKG04] VIOLA I., KANITSAR A., GRÖLLER M. E.: Importance-driven volume rendering. In *VIS '04: Proceedings of the conference on Visualization '04* (Washington, DC, USA, 2004), IEEE Computer Society, pp. 139–146.

[vWN04] VAN WIJK J. J., NUIJ W. A.: A model for smooth viewing and navigation of large 2d information spaces. *IEEE Transactions on Visualization and Computer Graphics 10* (2004), 447–458.

[WS07] WOODRING J., SHEN H.-W.: Incorporating highlighting animations into static visualizations. In *Proceedings of SPIE* (2007), vol. 6495.