Previewing Volume Decomposition Through Optimal Viewpoints*

Shigeo Takahashi[1], Issei Fujishiro[2], Yuriko Takeshima[3], and Chongke Bi[1]

1 **Graduate School of Frontier Sciences, The University of Tokyo**
5-1-5 Kashiwanoha, Kashiwa, Chiba 277-8561, Japan
takahashis@acm.org, bichongke@visual.k.u-tokyo.ac.jp

2 **Department of Information and Computer Science, Keio University**
3-14-1 Hiyoshi, Kohoku-ku, Yokohama 223-8522, Japan
fuji@ics.keio.ac.jp

3 **Institute of Fluid Science, Tohoku University**
2-1-1 Katahira, Aoba-ku, Sendai 980-8577, Japan
takesima@vis.ifs.tohoku.ac.jp

Abstract

Understanding a volume dataset through a 2D display is a complex task because it usually contains multi-layered inner structures that inevitably cause undesirable overlaps when projected onto the display. This requires us to identify feature subvolumes embedded in the given volume and then visualize them on the display so that we can clarify their relative positions. This article therefore introduces a new feature-driven approach to previewing volumes that respects both the 3D nested structures of the feature subvolumes and their 2D arrangement in the projection by minimizing their occlusions. The associated process begins with tracking the topological transitions of isosurfaces with respect to the scalar field, in order to decompose the given volume dataset into feature components called interval volumes while extracting their nested structures. The volume dataset is then projected from the optimal viewpoint that archives the best balanced visibility of the decomposed components. The position of the optimal viewpoint is updated each time when we peel off an outer component with our interface by calculating the sum of the viewpoint optimality values for the remaining components. Several previewing examples are demonstrated to illustrate that the present approach can offer an effective means of traversing volumetric inner structures both in an interactive and automatic fashion with the interface.

1998 ACM Subject Classification I.3.8 [Computer Graphics]: Applications

Keywords and phrases Interval volumes, viewpoint selection, feature-driven approach, volume peeling, nested structures

Digital Object Identifier 10.4230/DFU.Vol2.SciViz.2011.346

1 Introduction

The most common media for our visual communication have been 2D projections such as photographs, paintings, and computer displays. While the 2D projections are no doubt powerful means of conveying visual information, they still suffer from the inherent limitation on the dimensionality of the space where the information is presented, especially in the

* This work has been partially supported by Japan Society of the Promotion of Science under Grants-in-Aid for Young Scientists (B) No. 17700092, and the Ohkawa Foundation.

Scientific Visualization: Interactions, Features, Metaphors. *Dagstuhl Follow-Ups, Vol. 2.*
Editor: Hans Hagen; pp. 346–359

Dagstuhl Publishing
Schloss Dagstuhl – Leibniz Zentrum für, Germany

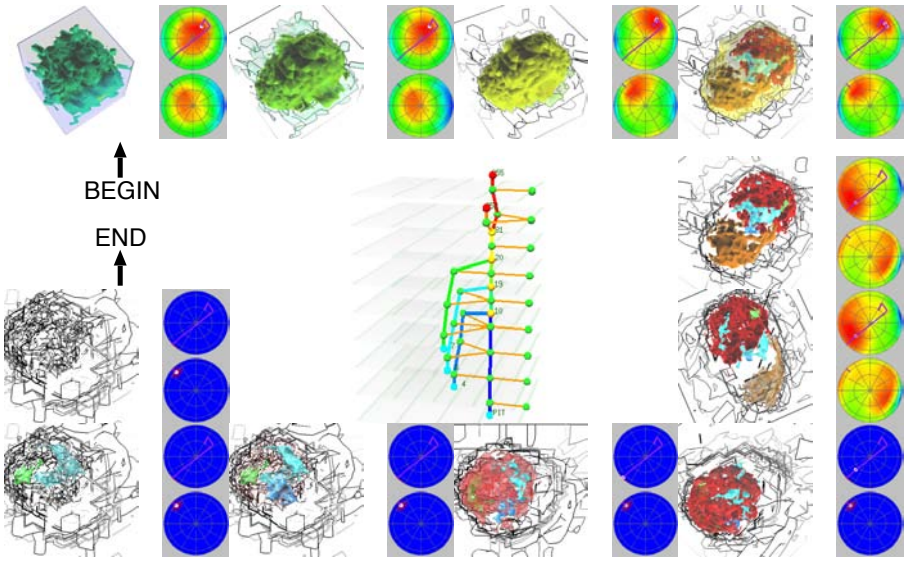

Figure 1 Scenario for decomposing the volume of "sheep heart". Each frame consists of the snapshot of volume decomposition preview and its associated map of viewpoint entropy. The graph at the center shows the corresponding interval volume structure.

case of 3D solid volumes. This is due to the fact that the volume datasets usually contain multi-layered inner structures where specific features cannot be easily identified through simple observations from the outside. Even if the associated features are identified, they cannot be projected clearly onto the 2D image plane because they are usually accompanied with undesirable overlaps while transparent rendering alleviates the problem to a certain extent.

This article therefore presents an approach for previewing volume inner structures by respecting both their 3D nested structures in the volume and their 2D arrangements in the projection. The approach consists of two ingredients: an algorithm for extracting feature components from the given volume so that we can peer into the volume by removing an outer component one by one, and an algorithm for locating an optimal viewpoint that provides the best arrangement of the feature components on the 2D projection. Our contribution lies in the combination of these two feature-driven algorithms because this is, to our knowledge, the first attempt to respect the underlying feature components consistently in both analyzing and previewing the inner structures of the volume.

The first algorithm has been implemented in our previous study as an interface called *interval volume decomposer* [28]. The interface decomposes an entire volume into feature subvolumes by tracking the topological transitions of the corresponding isosurface with respect to the scalar field, and then by constructing a graph structure called *a level-set graph* that represents the topological skeleton of the given volume. Furthermore, the link of the level-set graph corresponds to a feature subvolume called *an interval volume* [10, 13] in this framework. This enables systematic decomposition of the given volume from outside to inside by referring to the constructed level-set graph.

The second algorithm has been implemented in our previous formulation of viewpoint entropy for volumes [29]. Actually, this formation takes account of the combinations of feature interval volumes and evaluates the optimality of their arrangements in 2D projections. In each decomposition step, we can smoothly change the associated viewpoint by updating

the map of the entropy values over the viewing sphere.

Fig. 1 shows an example where the inner structure of a sheep heart dataset [23] is effectively presented by incorporating the above two algorithms into our previewing interface. As shown in the figure, the decomposed feature subvolumes are rendered in different colors and the outermost subvolume is faded out one by one while its silhouette gradually appears. Every time one subvolume is taken away, the entropy map over the viewing sphere together with the position of the next optimal viewpoint is updated. In Fig. 1, the upper and lower halves of the viewing sphere are rendered with the color-coded distribution of the viewpoint entropy, where warm colors represent higher entropy regions. Note that the purple line indicates the viewpoint path over the viewing sphere through the entire preview of the decomposition process.

This article is organized as follows: Section 2 surveys previous studies related to the present approach. Section 3 describes an algorithm for decomposing an entire volume into feature subvolumes by referring to its associated level-set graph. Section 4 explains a formulation of the viewpoint entropy for volumes by taking account of the characteristic combinations of the decomposed subvolumes. An interface for systematically decomposing a given volume dataset together with its optimal viewpoint updates is presented in Section 5. Section 6 concludes this article and refers to our future extensions.

2 Related Work

Our approach to volume previewing is related to several areas of research in visualization, computer graphics, and computer vision. Among the research areas, this section gives brief surveys on interactive volume exploration and optimal viewpoint selection.

2.1 Interactive Volume Exploration

Exploring the inside of a volume interactively makes it possible to effectively understand its underlying complex structure. *Volume sculpting* techniques are among such approaches where users can analyze the multi-layered inner structure with interactive carving and sawing operations [11, 36]. Volume peeling can be thought of as one of the principal editing processes used in the volume sculpting systems.

On the other hand, an idea of *haptic rendering* has emerged as the virtual reality technologies have been developed. Actually, the haptic rendering is a process of generating forces and torques in response to user interactions with virtual objects through haptic devices [15], and has also been introduced to volume exploration systems [1]. Anatomical metaphors for medial volumes obtained by CT and MRI scans have also inspired many methods for exploring volume datasets [19, 20]. In addition to these software-based approaches, hardware-assisted volume deformation techniques have also been developed [18, 38].

Another interesting approach is to preform the rigorous analysis of the given volume prior to the volume exploration. Pioneering work was done by Bajaj et al. [2], where they used contour trees to effectively explore the volume inside. Here, the contour tree represents a level-set graph of the given volume, which delineates the topological transitions of an isosurface according to the change of the scalar field value as shown in Fig. 2. Thus the contour trees have been intensively used to as a tool for exploring the isosurface trajectories in subsequent studies [5, 31, 37] while the associated algorithms for computing the level-set graphs have been also developed [34, 22, 6].

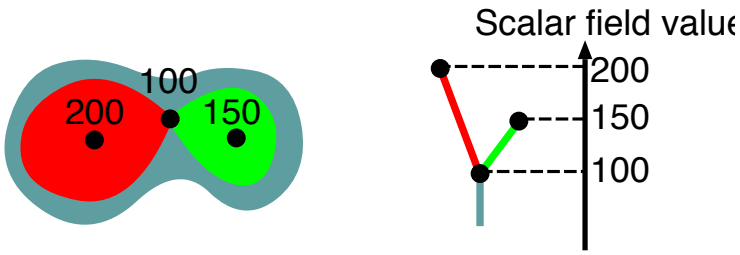

Figure 2 Isosurface transitions (on the left) and the corresponding level-set graph (on the right). The nodes of the level-set graph represent critical points in a volume, and are arranged according to their scalar field values in this article. The same color is assigned to a link of the level-set graph (on the right) and its corresponding subvolume (on the left).

2.2 Viewpoint Selection

The viewpoint selection problem has been studied rather by researchers in computer vision. One of such pioneering studies on this problem was conducted by Koenderink et al. [17], where they defined an aspect graph that partitions the viewing sphere that surrounds the target object into *aspects* by identifying equivalent views of the object edges in a topological sense. The aspect graph representation has been intensively studied as a tool for object recognition [7], while its automatic computation is still the subject of ongoing research [25]. On the other hand, finding a set of optimal camera positions requires some criteria for evaluating the goodness of each classified view position. This problem of planning camera positions is referred to as the *next best view (NBV)* problem in the computer vision literature, and has been investigated by a number of researchers, for example in [12, 24]. The configuration of such optimal viewpoints is also useful for acquiring a minimal set of textures in the image synthesis techniques such as image-based rendering [8, 14].

In the computer graphics applications, several methods have been proposed to locate the single best viewpoint for 3D polyhedral meshes [16, 3]. Among these methods, the most reliable solution especially for the case of general 3D meshes is the *viewpoint entropy*, which is formulated by Vázquez et al. [35] to evaluate the balance of visible faces in 2D projected images.

Locating optimal viewpoints for volumes apparently presents another problem because the volumes usually involve multi-layered inner structures and can be thought of as one-dimension higher objects than surface meshes. Recently, this problem was explored simultaneously but independently by Bordoloi et al. [4] and Takahashi et al. [29]. Bordoloi et al. evaluated the balance between the contributions of voxels to pixels in the resultant image, while Takahashi et al. tried to find the well-balanced arrangement of feature subvolumes on the 2D projected image.

3 Interval Volume Decomposer

This section describes an algorithm that tracks the level-set graph for systematically decomposing a given volume into a set of feature subvolumes, followed by a framework for previewing the volume in our interface called *interval volume decomposer* [28].

3.1 Interval Volumes

For extracting the feature subvolumes from the input dataset, our algorithm first constructs the level-set graph that tracks the topological transitions of an isosurface according to the scalar field, and then identifies each link of the graph with a feature subvolume. Actually, in this case, the subvolume corresponds to an interval volume (IV) bounded by critical isosurfaces. The concept of IV is formulated by Fujishiro et al. [10] and Guo [13] as a generalization of the isosurface, and defined as a subvolume that corresponds to some range of the scalar field value. Our framework employs a set of such IVs obtained by referring to the constructed level-set graph, where the IVs also play an important role in selecting the optimal viewpoint for seeking the best of their arrangements in 2D projection (Section 4.2).

3.2 Level-set Graph Construction

For constructing the level-set graph, we have developed an algorithm called *topological volume skeletonization* [31], which tracks the topological transitions of an isosurface according to the scalar field. Actually, the algorithm is constructed by combining the algorithm of Carr et al. [6] for tracking the number of isosurface connected components, the algorithm of Pascucci et al. [22] for identifying the genus of each isosurface component, and the algorithm of Takahashi et al. [31] for IV clustering. Refer to [30] for the practical implementation of this algorithm.

3.3 Isosurface Inclusion Relationships at Saddles

The constructed level-set graph illuminates a systematic decomposition of the given volume by relating its links to the decomposed feature IVs. This implies that peeling the decomposed IVs from outside to inside requires to locate their inclusion relationships. According to [32], such inclusion relationships can be retrieved by systematically traversing the level-set graph from the end node that corresponds to the exterior surface of the given volume.

A node of the level-set graph represents a critical point at which an isosurface component has topological evolution. Mathematically, critical points in volumes are classified into C_3 (maxima), C_2 (saddles), C_1 (saddles), and C_0 (minima) according to their indices, where the index represents the number of negative eigenvalues of the corresponding Hessian matrix. Taking account of the isosurface embeddings in 3D space allows us to classify the isosurface transitions at C_2 and C_1 into four different isosurface transitions as shown in Fig. 3 [31]. Here, we call an isosurface *solid* if it expands as the scalar field value decreases while *hollow* if it shrinks. The leftmost and rightmost columns in Fig. 3 illustrate subgraphs around the critical points of C_1 and C_2 where a different color is assigned to each link according to whether the corresponding isosurface is solid or hollow. This figure suggests that IV inclusions occur only in the two transition paths of the row (B) for both C_2 and C_1.

3.4 Determining the IV Decomposition Order

In the present algorithm, the IV decomposition order is extracted by tracing the links of the level-set graph from the end node that represents the exterior boundary of the given volume. Fig. 4 shows an example where we determine the decomposition order by tracing the level-set graph. Note that the associated algorithm prepares two FIFO queues Q_{solid} and Q_{hollow} for solid and hollow links, respectively, and adds links traversed in the upward direction to Q_{solid} while those traversed in the downward direction to Q_{hollow}. In addition,

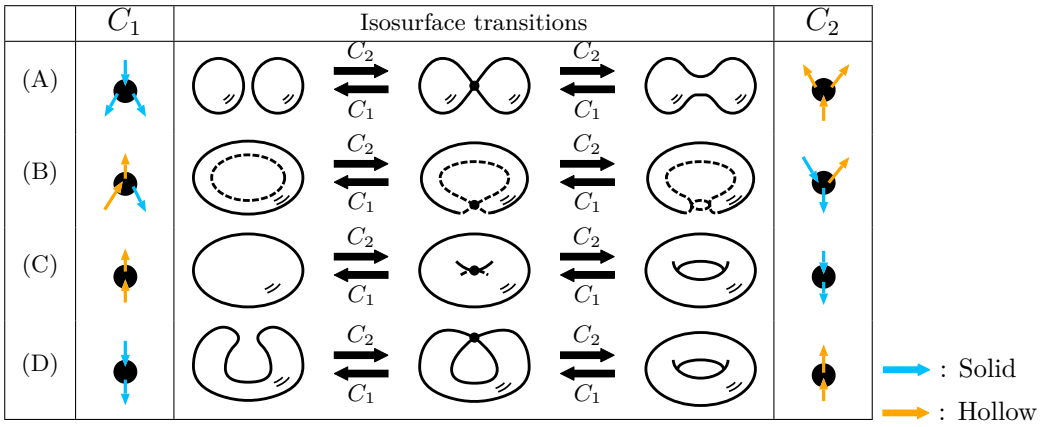

Figure 3 Classification of isosurface transitions at saddle critical points of C_2(join) and C_1(split) according to the spatial configuration in 3D space. The horizontal arrows in the central column indicate isosurface transitions as the scalar field value reduces. Different colors are assigned to solid and hollow links in the level-set graph.

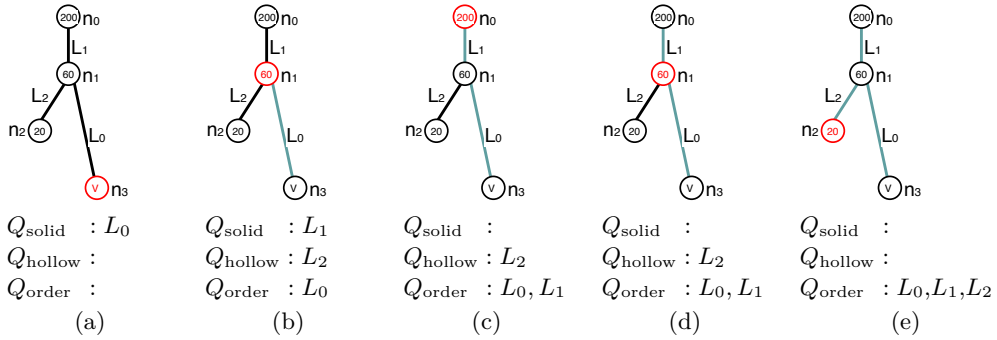

Figure 4 Steps for extracting the IV decomposition order.

Q_{order} is introduced to retain the links that represent the IV decomposition order. Initially, Q_{hollow} and Q_{order} are empty while Q_{solid} has the link L_0 to be examined first (Fig. 4(a)).

The actual tracing process starts with the traversal from the node n_3 to n_1 through the link L_0 as shown in Fig. 4(a), where L_0 is solid because the it is connected to the exterior surface of the given volume. This lets us delete the link L_0 from Q_{solid} and add it to Q_{order}. Since the associated algorithm identifies the node n_1 as (B)-C_1 type according to the classification in Fig. 3, it inserts L_1 to Q_{solid} and L_2 to Q_{hollow} (Fig. 4(b)). The upward tracing process continues to handle all the links in Q_{solid} until it becomes empty (Fig. 4(c)). The algorithm then begins to handle the links in Q_{hollow}. In this case, it resumes the traversal from n_1 to n_2 through L_2 in the downward direction (Fig. 4(d)). Finally, the tracing process is completed by storing the decomposition order of IVs in Q_{order} while identifying the inclusion relationship between L_0 and L_2 (Fig. 4(e)).

4 Viewpoint Entropy for Volumes

Viewpoint selection for previewing volumes needs criteria for calculating viewpoint optimality for the given volume. The present algorithm calculates such viewpoint goodness by evaluating

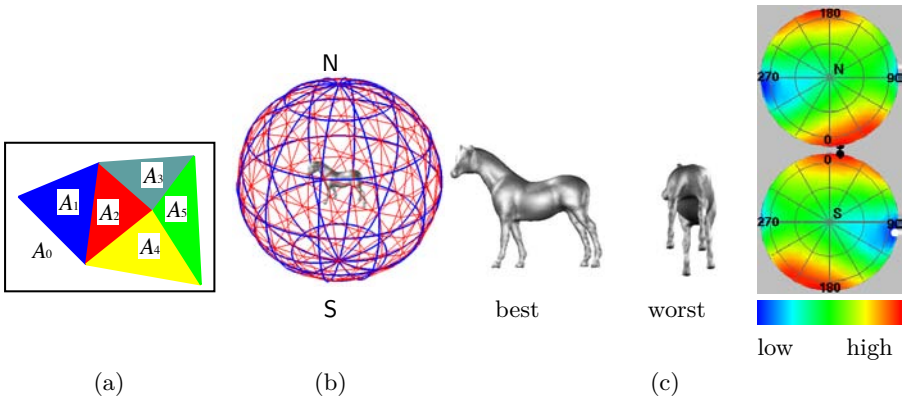

Figure 5 (a) Assignment of different colors to visible faces on the 2D screen. (b) A reference mesh (in red) on the viewing sphere (in blue) for viewpoint samples. (c) The viewpoint entropy distributions and the associated best and worst views for the horse model.

the arrangement of decomposed IVs in the 2D projection using the conventional surface-based technique.

4.1 Formulation of the Viewpoint Entropy for Surfaces

As the criteria for evaluating optimal views of surface meshes, we employ the *viewpoint entropy* formulated by Vázquez et al. [35], which searches for a well-balanced distribution of visible faces using the entropy measure. In our algorithm, we modified the original formulation of the viewpoint entropy so that we can evaluate the optimality of view directions under orthographic projections. Suppose that the j-th face of a given 3D mesh has a visible projected area A_j $(j = 1, \ldots, m)$ on the 2D screen, while A_0 denotes the background area as shown in Fig. 5(a). Thus, the total area of the 2D screen S can be calculated as $S = \sum_{j=0}^{m} A_j$. The normalized version of the viewpoint entropy E can be defined as

$$E = -\frac{1}{\log_2(m+1)} \sum_{j=0}^{m} \frac{A_j}{S} \log_2 \frac{A_j}{S}. \tag{1}$$

Note that the quantity E in Equation (1) becomes larger as the corresponding viewpoint achieves more balanced distribution of face visibility, while it ranges from 0.0 (when all the faces are invisible) to 1.0 (when all the faces share the same area).

The actual calculation of the viewpoint entropy in Equation (1) is carried out as follows [3]: Given a viewpoint, the projected area of each face of the target mesh is obtained by counting the number of pixels that belong to the face. Assigning a different color to each face helps us discriminate one face from the others as shown in Fig. 5(a). The associated viewpoint entropy is evaluated at vertices of another reference mesh (in red) that covers the viewing sphere (in blue) as shown in Fig. 5(b). Fig. 5(c) shows the distributions of the viewpoint entropies in Equation (1) over the viewing sphere, together with the associated best and worst views of the 3D horse model, where the best viewpoint captures its silhouettes clearly. The associated top and bottom disks are the projections of the viewpoint entropy distributions on the viewing sphere seen from the top (the North pole) and the bottom (the South pole), respectively. Here, each disk is color-coded by referring to the color legend where the color changes over blue, green, yellow, and red according to the increase in the entropy, and the black and white spots indicate the best and worst locations of the viewpoints, respectively.

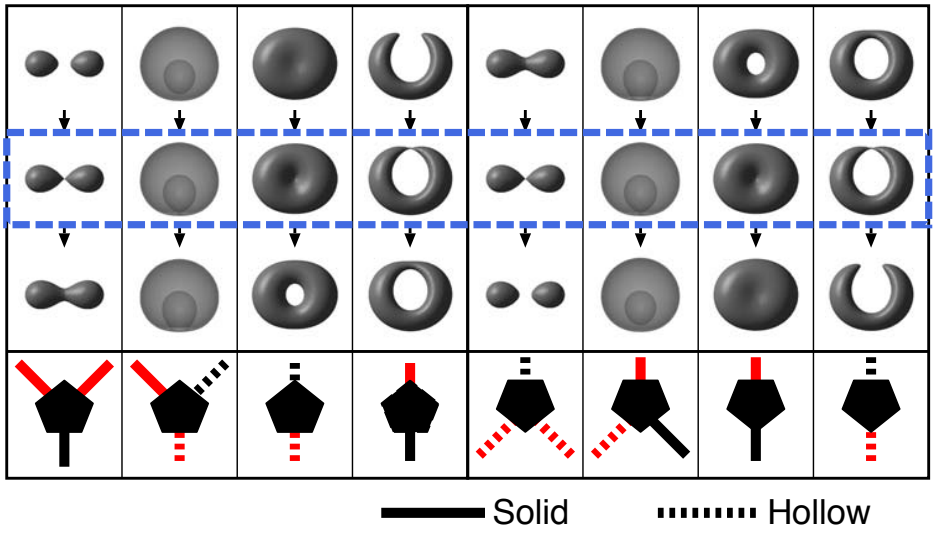

Figure 6 The combinations of IVs for the viewpoint evaluation and their associated links in level-set graphs.

4.2 Formulation of the Viewpoint Entropy for Volumes

For locating the optimal viewpoint for volumes, we introduce our feature-driven approach [29] where we calculate the viewpoint entropies of feature components and then find the global compromise between them, by taking advantage of the aforementioned surface-based entropy formulation. As the feature components, our approach employs combinations of the decomposed feature IVs, each of which constitutes an isosurface join or split of some specific type. Fig. 6 depicts such combinations of IVs where the corresponding links are drawn in red. Note that the figure depicts subgraphs around saddle critical points in the level-set graph when the links are arranged from top to bottom with respect to the scalar field.

The viewpoint entropy for volumes is formulated as follows: Suppose that we separately evaluate each IV of some specific combination as shown in Fig. 6. The viewpoint entropy for the IV can be calculated using Equation (1) by identifying the visible faces of the IV. However, if the faces of the IV are partially occluded by the other IV in the combination, the occluded regions are assumed be painted in the background color when evaluating the entropy. This handling together with Equation (1) allows us to avoid undesirable occlusions between the IVs in the combination. For evaluating the globally optimal viewpoint, our algorithm calculates the weighted sum of the viewpoint entropies, which is given as

$$E = \sum_{i=1}^{n} \frac{\lambda_i}{L} E_i, \tag{2}$$

by calculating Equation (1) for the i-th IV as E_i. Here, λ_i is the weight value for the i-th IV and $L = \sum_{i=1}^{n} \lambda_i$. In addtition, n is the number of IV. In our framework, λ_i is calculated using an opacity transfer function for direct volume rendering, as follows:

$$\lambda_i = \frac{1}{M_i} \sum_{j=1}^{M_i} o(\boldsymbol{x}_j) \tag{3}$$

where $o(\boldsymbol{x}_j)$ is an opacity value at the j-th voxel sample $\boldsymbol{x}_j$, and M_i is the number of voxel samples $\boldsymbol{x}_j$ contained in the i-th IV. Here, the opacity transfer function is used to assign large

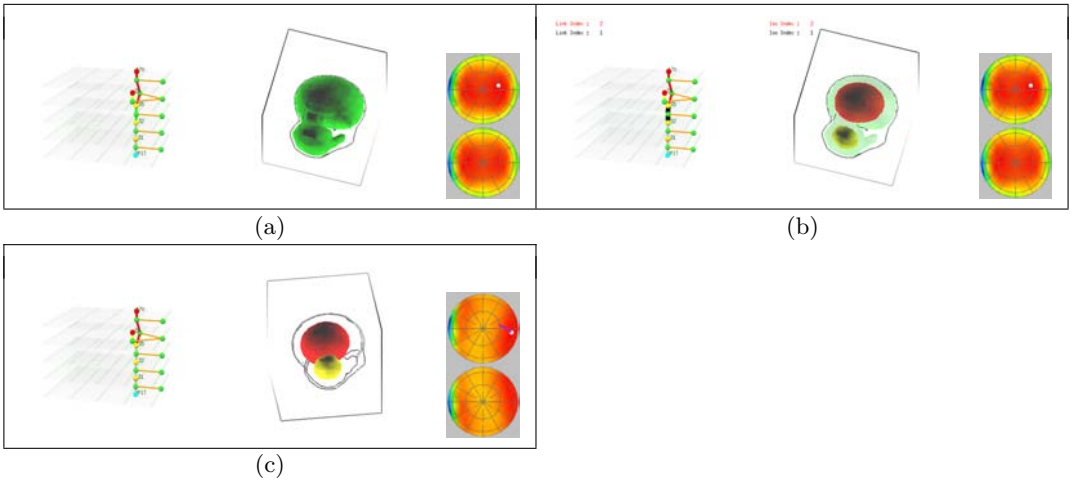

Figure 7 Screen snapshots of the interface where IVs are peeled off interactively: (a) The outermost IV (in green) is specified by a pointing device. (b) The selected IV becomes transparent (in the middle) while its corresponding link is emphasized (on the left). (c) After the selection is confirmed, the IV is removed and its silhouettes are displayed in place (in the middle) while the corresponding link is faded (on the left). The entropy distribution is updated and the viewpoint moves to the next optimal position along the purple path (on the right).

weights to the feature IVs we want to emphasize. Needless to say, more sophisticated transfer functions, such as multi-dimensional transfer functions [33] can be specified in Equation (3), without any modifications to the remainder of the present framework.

5 Interface for Previewing Volume Decomposition

This section describes the implementation of our interface that systematically decomposes the given volume from outside to inside while tracking the movement of its optimal viewpoint over the viewing sphere. The remainder of this section is devoted to describing how to accomplish the IV decomposition using the present interface, either interactively (Section 5.1) or automatically (Section 5.2).

5.1 Interactive IV Decomposition

The interactive IV decomposition proceeds by specifying a feature IV to be excluded using a pointing device such as a mouse. Fig. 7 shows screen snapshots of our interface where it previews the proton atom-hydrogen atom collision dataset [9]. The leftmost window is for displaying the level-set graph together with its IV inclusion relationships. The middle window is for presenting the decomposed feature IVs where a user can peel off IVs from outside to inside one by one. The rightmost window is for exhibiting the color-coded distribution of the viewpoint entropy calculated from the remaining combinations of feature IVs. The two left windows provide an interface for selecting an individual IV using a pointing device while the rightmost window relates the smooth movement of the viewpoint. In addition, the same color is assigned to the link of the level-set graph and its corresponding feature IV in order to visualize the correspondence between them.

As shown in Fig. 7(a), a user specifies the outermost IV to be removed either by clicking

the green subvolume in the middle window or by clicking the green link in the left window. Furthermore, the interface indicates the position of the current optimal viewpoint by a black spot on the projection of the upper viewing hemisphere. Note that the distribution of the viewpoint entropy is calculated as the sum of the viewpoint entropies assigned to the remaining combinations of IVs. The interface changes the status of the selected IV by rendering it transparently in the middle window and emphasizing the corresponding link of the level-set graph in the left window, as shown in Fig. 7(b). At this point, the user can look at the structure of the interior IVs through the outer transparent one. When the user confirms this selection, the interface finally removes the selected IV and leaves its silhouettes instead in the middle window, while the corresponding link in the left window is grayed out, as shown in Fig. 7(c). In addition, the interface recalculates the distribution of the viewpoint entropy for the updated combinations of the remaining IVs in the right window, and changes the view of the decomposed IVs in the middle window so that we can smoothly move the corresponding viewpoint from the previous best viewpoint to the next one. The path of the viewpoint movement is obtained by the spherical linear interpolation based on quaternion representations [26]. As demonstrated in Fig. 7, the interface provides an effective means of eliminating decomposed IVs one by one from outside to inside together with the comfortable viewpoint changes, while allowing the user to fully understand the multi-layered inner structures of the target volume.

5.2 Automatic IV Decomposition

Furthermore, the interface automatically generates a scenario for peeling off the IVs from outside to inside using the IV decomposition order as an animation, while smoothly changing its viewpoint positions. Figs. 1, 8, and 9 present scenarios for volume peeling using the present interface. In these scenarios, the decomposed IVs gradually disappear one by one in accordance with the decomposition order by tracking the constructed level-set graph (Section 3). Furthermore, the silhouettes of the IVs will gradually become conspicuous as the corresponding IVs become transparent. Throughout the decomposition scenario, the interface changes the viewpoint position along the path that follows the optimal positions obtained in each step of eliminating the outermost IV.

As described in Section 1, Fig. 1 shows a scenario in which the volume of the sheep heart ($177 \times 177 \times 177$) [23] is dissected from outside to inside using the interface. This example demonstrates that the present interface works well and provides a systematic process of decomposing nested structures even for such an anatomical volume, together with the smooth change of the viewpoint indicated in purple in the figure. Fig. 8 visualizes a simulated dataset ($41 \times 41 \times 41$), where the two-body distribution probability of a nucleon in the atomic nucleus ^{16}O is computed [21]. This dataset contains a two-fold nested structure of IVs and reveals its attractive interior while the viewpoint movement along the purple path first provides its side view and then its top view. Fig. 9 represents another dataset ($129 \times 129 \times 129$), which is obtained by simulating the antiproton-hydrogen atom collision at intermediate collision energy [27]. Note that the interface successfully resolves the complicated structure of this dataset even though it contains a four-fold nested structure of IVs. The viewpoint path (in purple) effectively tracks the best views for the associated decomposition steps while it goes around through the optimal viewpoint positions scattered over the viewing sphere. These results demonstrate the potential and feasibility of the present framework.

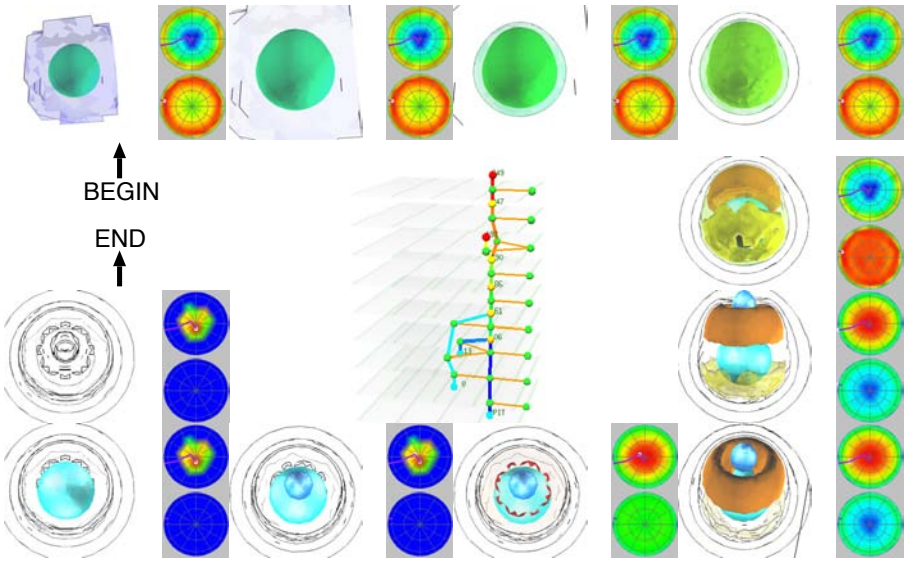

Figure 8 Scenario for decomposing the volume of "nucleon". Each frame consists of the snapshot of volume decomposition preview and its associated map of viewpoint entropy. The graph at the center shows the corresponding interval volume structure.

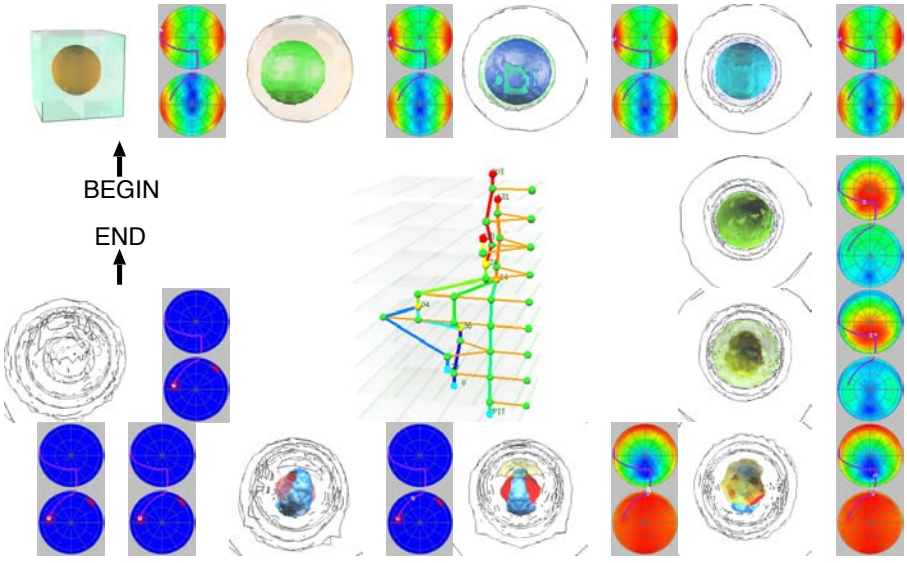

Figure 9 Scenario for decomposing the volume of "antiproton-hydrogen atom collision". Each frame consists of the snapshot of volume decomposition preview and its associated map of viewpoint entropy. The graph at the center shows the corresponding interval volume structure.

6 Conclusion and Future Work

This article has presented an approach to previewing volume datesets respecting both its 3D nested inner structures and their 2D arrangements in the projections. This is accomplished by first constructing the level-set graph that identifies the feature subvolumes by tracking the topological transitions of an isosurface, and then projecting them onto 2D screen so that their arrangements become optimal with less occlusions. Interval volumes (IVs) are introduced as the feature subvolumes where each IV corresponds to a link of the level-set graph, which makes it possible to investigate nested relationships between the decomposed IVs. An algorithm for locating the optimal viewpoint for volumes is also presented that tries to minimize the occlusions between feature interval volumes by taking account of their characteristic combinations inheriting from the level-set graph. Implementation of the associated interface together with several previewing examples is included to demonstrate the feasibility of the present approach.

Future extensions include improving the interface so that it can provide multiple views of the target volumes so that the corresponding view frustums effectively cover the entire 3D domain where the target volume is defined. Currently we use the transfer function to assign a large weight to some specific interval volume. However, we can interactively assign specific weights to the interval volumes with the help of interface and confirm its validity by calculating the corresponding viewpoint position. Furthermore, editing several effects such as coloring, lighting, and rendering styles with the interface is one of the interesting future themes.

References

1 R. S. Avila and L. M. Sobierajski. A haptic interaction method for volume visualization. In *Proc. of IEEE Visualization '96*, pages 197–205,485, 1996.

2 C. L. Bajaj, V. Pascucci, and D. R. Schikore. The contour spectrum. In *Proc. of IEEE Visualzation '97*, pages 167–173, 1997.

3 P. Barral, G. Dorme, and D. Plemenos. Scene understanding techniques using a virtual camera. In *Proc. of Eurographics '00 Short Presentations*, 2000.

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

5 H. Carr and J. Snoeyink. Path seeds and flexible isosurfaces using topology for exploratory visualization. In *Proc. of Joint Eurographics-IEEE TCVG Symp. on Visualization*, pages 49–58, 285, 2003.

6 H. Carr, J. Snoeyink, and U. Axen. Computing contour trees in all dimensions. *Computational Geometry*, 24(2):75–94, 2003.

7 C. M. Cyr and B. B. Kimia. 3D object recognition using shape similarity-based aspect graph. In *Proc. of Int. Conference on Computer Vision*, pages 254–261, 2001.

8 S. Fleishman, D. Cohen-Or, and D. Lischinski. Automatic camera placement for image-based modeling. *Computer Graphics Forum*, 19(2):101–110, 2000.

9 I. Fujishiro, T. Azuma, Y. Takeshima, and S. Takahashi. Volume data mining using 3D field topology analysis. *IEEE CG&A*, 20(5):46–51, 2000.

10 I. Fujishiro, Y. Maeda, H. Sato, and Y. Takeshima. Volumetric data exploration using interval volume. *IEEE Trans. on Visualization and Computer Graphics*, 2(2):144–155, 1996.

11 T. A. Galyean and J. F. Hughes. Sculpting: An interactive volumetric modeling technique. In *Computer Graphics (Proc. of Siggraph '91)*, pages 267–274, 1991.

12 K. D. Gremban and K. Ikeuchi. Planning multiple observations for object recognition. *International Journal of Computer Vision*, 12:137–172, 1994.

13 B. Guo. Interval set: A volume rendering technique generalizing isosurface extraction. In *Proc. of IEEE Visualization '95*, pages 3–10, 1995.

14 Y. Iwakiri and T. Kaneko. PC-based real-time texture painting on real world objects. *Computer Graphics Forum*, 20(3):105–113, 2001.

15 H. Iwata and H. Noma. Volume haptization. In *Proc. of IEEE 1993 Symposium on Research Frontiers in Virtual Reality*, pages 16–23, 1993.

16 T. Kamada and S. Kawai. A simple method for computing general position in displaying three-dimensional objects. *Computer Vision, Graphics, and Image Processing*, 41(1):43–56, 1988.

17 J. J. Koenderink and A. J. van Doorn. The singularities of the visual mapping. *Biological Cybernetics*, 32:211–216, 1979.

18 Y. Kurzion and R. Yagel. Interactive space deformation with hardware-assisted rendering. *IEEE CG&A*, 17(5):66–77, 1997.

19 W. E. Lorensen. Geometric clipping using boolean textures. In *Proc. of IEEE Visualization '93*, pages 268–274, 1993.

20 M .J. McGuffin, L. Tancau, and R. Balakrishnan. Using deformations for browsing volume data. In *Proc. of IEEE Visualization 2003*, pages 401–408, 2003.

21 M. Meißner. Web Page [http://www.volvis.org/].

22 V. Pascucci and K. Cole-McLaughlin. Efficient computation of the topology of level sets. In *Proc. of IEEE Visualization 2002*, pages 187–194, 2002.

23 H. Pfister, et al. The transfer function bake-off. *IEEE CG&A*, 21(3):16–22, 2001.

24 R. Pito. A solution to the next best view problem for automated surface aquisition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(10):1016–1030, 1999.

25 I. Shimshoni and J. Ponce. Finite-resolution aspect graphs of polyhedral objects. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(4):315–327, 1997.

26 K. Shoemake. Animating rotation with quaternion curves. In *Proc. of Siggraph '85*, pages 245–254, 1985.

27 R. Suzuki, H. Sato, and M. Kimura. Antiproton-hydrogen atom collision at intermediate energy. *IEEE Computing in Science and Engineering*, 4(6):24–33, 2002.

28 S. Takahashi, I. Fujishiro, and Y. Takeshima. Interval volume decomposer: A topological approach to volume traversal. In *Proc. of SPIE Conference on Visualization and Data Analysis 2005*, volume 5669, pages 103–114, 2005.

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

30 S. Takahashi, G. M. Nielson, Y. Takeshima, and I. Fujishiro. Topological volume skeletonization using adaptive tetrahedralization. In *Proc. of Geometric Modeling and Processing 2004*, pages 227–236, 2004.

31 S. Takahashi, Y. Takeshima, and I. Fujishiro. Topological volume skeletonization and its application to transfer function design. *Graphical Models*, 66(1):22–49, 2004.

32 S. Takahashi, Y. Takeshima, I. Fujishiro, and G. M. Nielson. Emphasizing isosurface embeddings in direct volume rendering. In G.-P. Bonneau, T. Ertl, and G. M. Nielson, editors, *Scientific Visualization: The Visual Extraction of Knowledge from Data*, pages 185–206. Springer-Verlag, 2005.

33 Y. Takeshima, S. Takahashi, I. Fujishiro, and G. M. Nielson. Introducing topological attributes for objective-based visualization of simulated datasets. In *Proc. of Volume Graphics 2005*, pages 137–145, 236, 2005.

34 M. van Kreveld, R. van Oostrum, C. Bajaj, V. Pascucci, and D. Schikore. Contour trees and small seed sets for isosurface traversal. In *Proc. of 13th ACM Symp. on Computational Geometry*, pages 212–220, 1997.

35 P.-P. Vázquez, M. Feixas, M.Sbert, and W. Heidrich. Viewpoint selection using view entropy. In *Proc. of Vision Modeling and Visualization Conference (VMV01)*, pages 273–280, 2001.

36 S. Wang and A. E. Kaufman. Volume sculpting. In *Proc. of the 1995 Symp. on Interactive 3D Graphics*, pages 151–156, 1995.

37 G. H. Weber and G. Scheuermann. Automating transfer function design based on topology analysis. In *Geometric Modeling for Scientific Visualization*, pages 293–306. Springer-Verlag, 2004.

38 D. Wiskopf, K. Engel, and T. Ertl. Volume clipping via per-fragment operations in texture-based volume visualization. In *Proc. of IEEE Visualization 2002*, pages 93–100, 2002.