Adjusting Control Parameters of Topology-Accentuated Transfer Functions for Volume Raycasting

Yuriko Takeshima, Shigeo Takahashi, and Issei Fujishiro

Abstract The design of automatic transfer functions for volume rendering is a perennial problem in volume visualization. Over the last three decades, a variety of design methodologies have been proposed. However, sensitive adjustment of related control parameters remains entrusted to users, because rendering conditions, such as the thickness of emphasized subvolumes in the ray direction and the size of a target dataset, differ on a case-by-case basis. Our group previously proposed one-dimensional transfer functions to accentuate topological changes in the scalar field of the target dataset. However, the method forces us to determine the actual control parameter values for the transfer functions in an empirical manner. In this paper, we propose a supplementary mechanism with which to judiciously define an appropriate profile of the opacity values. More specifically, the height and width of the hat opacity transfer functions that accentuate feature isosurfaces are determined according to the number of voxels belonging to the relevant topologically equivalent scalar field interval. The feasibility of the proposed method is evaluated by its application to five kinds of volume datasets.

1 Introduction

Volume rendering is one of the traditional visualization techniques for scalar fields. Over the last three decades, various methodologies have been proposed for designing transfer functions [1]. Data-centric approaches, which perform a

---

Y. Takeshima (✉)
Tokyo University of Technology, Tokyo, Japan
e-mail: takeshimayrk@stf.teu.ac.jp

S. Takahashi
University of Aizu, Fukushima, Japan
e-mail: takahashis@acm.org

I. Fujishiro
Keio University, Yokohama, Japan
e-mail: fuji@fj.ics.keio.ac.jp

mathematical analysis of the dataset before pertinent rendering, have become well established. Kindlmann and Durkin [2] defined a histogram volume consisting of first and second partial derivatives of the volume dataset, which is used to design transfer functions in order to emphasize the boundaries between different materials contained in the target volume dataset. Kniss et al. [3] generalized the histogram volume-based method in order to design three-dimensional transfer functions. Hladůvka et al. [4] and Kindlmann et al. [5] independently proposed multi-dimensional transfer functions based on isosurface curvatures in the target volume dataset. Weber et al. [6] proposed a topology-based method that defines the opacity at a sample point based on the topological characterization of a target dataset. Zhou et al. [7] proposed an automatic transfer function design using a residue flow model controlled by a contour tree [8]. Shape-based transfer functions, proposed by Praßni et al. [9], are based on the shape of the surfaces to be visualized. Xiang et al. [10] proposed a graph cut segmentation method and localized transfer function, which is suitable for datasets with a comparatively clear boundary rather than for simulation datasets whose values are distributed smoothly. In addition, it is conceivable to design the transfer functions based on other structures of the target dataset, such as representative isosurfaces [11].

Our group previously proposed yet another topology-based method [12], which designs one-dimensional transfer functions to accentuate topological changes in the scalar field of the target dataset. However, the method forces us to determine the actual control parameter values for the transfer functions in an empirical manner; the users have to adjust the control parameter values according to a target dataset. In this paper, therefore, we propose a supplementary mechanism with which to judiciously define an appropriate profile of opacity values. More specifically, the height and width of the hat opacity transfer functions that accentuate the feature isosurfaces are determined according to the number of voxels belonging to the relevant topologically equivalent scalar field interval. We focus on a traditional one-dimensional transfer function that assigns an opacity to a field value and a multi-dimensional transfer function that assigns an opacity to a field value as well as other attributes, such us inclusion level [13]. To evaluate the proposed method, we use our transfer functions to visualize five different kinds of volume datasets. Furthermore, we introduce an evaluation mechanism for measuring the quality of the volume-rendered images with the designed transfer functions and also discuss the processing speed of our method.

2 Volume Skeletonization

First, we extract the topological structure from a target dataset using the topological volume skeletonization algorithm [12, 13].

2.1   Volume Skeleton Tree

The level-set graph, known as the volume skeleton tree (VST), allows us to evaluate the topological attributes of each voxel by illuminating both the global and local features of the volume dataset. A node of the VST represents a critical point that displays a change either in the number of connected isosurface components or in the genus of each of the isosurface components. Critical points are classified into four groups: maxima (C_3), saddles (C_2, C_1), and minima (C_0), which represent isosurface appearance, merging, splitting, and disappearance, respectively, as the scalar field value decreases. A link of the VST represents a topology-preserving connected component of interval volume (transition of isosurface) [14]. A link is defined as *solid* if its isosurface component expands as the scalar field value decreases; it is defined as *hollow* if it shrinks.

Both the isosurfaces merging at C_2 and splitting at C_1 have four topological transition paths with different isosurface spatial configurations, as shown in Fig. 1. In what follows, the VST uses the notation for the critical points with its own connectivity, as indicated in Fig. 1, where the solid incident link represents a solid isosurface and the broken link a hollow isosurface. The saddle points of $C_i\,(i = 1, 2)$ are classified into 3-C_i and 2-C_i, according to their degree (valence). For later convenience, all the boundary voxels are assumed to be connected to the virtual minimum having $-\infty$ as its scalar field value [12]. Note that the link to the C_0 node is solid when the node is the virtual minimum, as shown in Fig. 1.
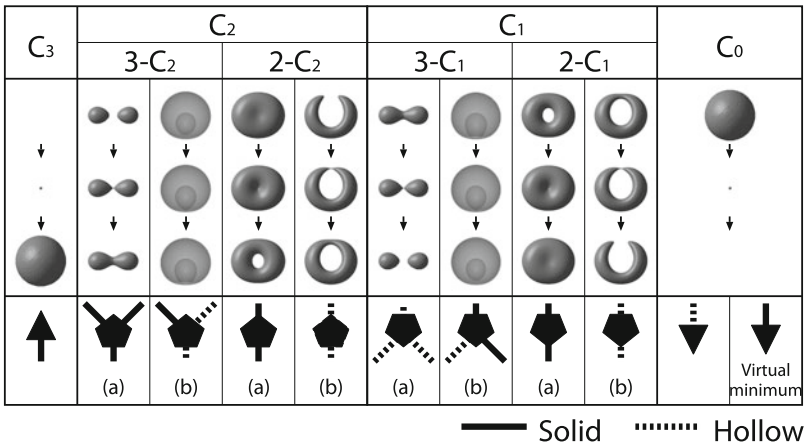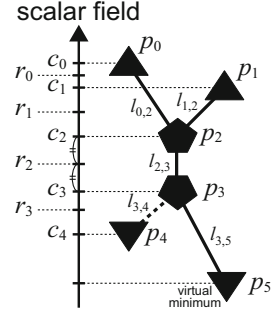

Fig. 1 The connectivity of critical points in the VST

Fig. 2 An example of a VST. The critical field values and the representative field values are denoted by c_i $(i = 0, \cdots, 4)$ and r_i $(i = 0, \cdots, 3)$, respectively. The inclusion level of the link $l_{3,4}$ is 1, whereas those of the outer links are 0

2.2 Feature Values in the VST

In a VST, a node p_i has coordinates $\boldsymbol{x}_i$ and a scalar field value c_i $(i = 0, \cdots, m-1)$, which is referred to as *critical field value*, denoted in the monotonically decreasing order of the scalar field. *Representative field value* is also defined as the mid-value $r_i = (c_{i+1} + c_i)/2$ of each interval $[c_{i+1}, c_i](i = 0, \ldots, m - 2)$ bounded by consecutive critical field values, as shown in Fig. 2.

A link $l_{i,j}$ has the genus of the corresponding isosurface component, the index of adjacent nodes p_i, and p_j, and its subvolume as the sweep of a connected component of a topologically equivalent isosurface. We denote a volume ratio belonging to the subvolume corresponding to link $l_{i,j}$ as $v_{i,j}$, calculated as $n_{i,j}/N$, where N denotes the total number of voxels in the target volume and $n_{i,j}$ denotes the number of voxels in the subvolume representing link $l_{i,j}$.

Because the VST is sensitive to small changes in field values, it may contain a large number of minor critical points if the volume dataset involves high-frequency noise. In order to capture the global features of the entire volume dataset, it is necessary to eliminate some minor links from the VST. To control the level of detail (LoD) of the VST, we define a persistency value of $l_{i,j}$ as $v_{i,j}/|c_i - c_j|$. In our algorithm, the VST is simplified by removing links in ascending order until the value reaches a specified threshold [12].

Furthermore, we can extract an inclusion level that represents the depth of its associated isosurface in the nested structure at the corresponding scalar field value and serves as an additional variable for the multi-dimensional transfer functions that emphasize these nested structures. Figure 1 clearly illustrates that isosurface nested structures originate only from the transition paths in 3-C_2(b) and 3-C_1(b). This motivates us to locate such isosurface inclusions directly from the VST if we can identify all the nodes that correspond to the previously mentioned transition paths. Indeed, the inclusion level can be systematically extracted by tracing the VST from the virtual minimum, because its incident link is known to be solid [13]. In Fig. 2, it has a nested structure in the field interval $[c_4, c_3]$ where the isosurfaces belonging to $l_{3,5}$ exist outside the isosurfaces belonging to $l_{3,4}$.

3   Transfer Function Design

Next, we design a transfer function reflecting the topological structures in the VST (Sect. 2). The basic principle is to accentuate the topological change of an isosurface within the volumetric domain [12].

3.1   Color Transfer Function

In our previously proposed method [13], wherein the color transfer function is defined piecewise using the range of the HSV hex-cone $[0, 2/3\pi]$, so that the function value decreases linearly over an evenly divided hue interval for each of the field intervals $[c_{i+1}, c_i]$. Our transfer functions can be designed based on blue-white-red, blue-white-yellow, heat object, as well as rainbow colormap. In every case, we define the color transfer function based on the above principle of dividing a given hue range. Figure 3a shows the design of our color transfer function, which allows us to assign a steep color gradation to the regions where the consecutive critical field values are in closer proximity to one another.

3.2   One-Dimensional Opacity Transfer Function

For opacity transfer functions, we propose two methods, accentuating critical or representative field values. The first method assigns local hat functions centered at the critical field values c_i, whereas the second is centered at representative field values r_i. When the critical field values are emphasized, topological changes, such as splitting and merging isosurfaces, can be visualized more clearly. On the other hand, emphasizing the representative field values reveals the representative topological structures. In our framework, because the outermost isosurface does not shrink as the field value decreases, we minimize the occlusion artifacts induced by the isosurface nested structure by decreasing the base elevation of the hat functions
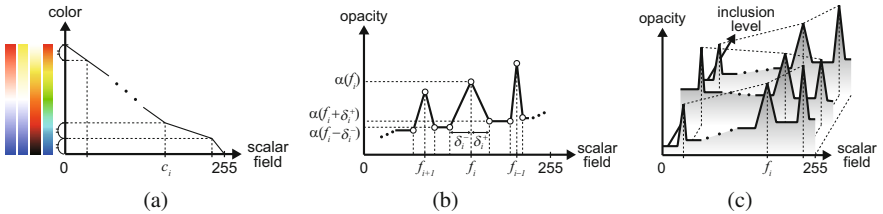

Fig. 3 Basic principle of designing transfer functions (TFs). (**a**) Color TF. (**b**) 1D opacity TF. (**c**) 2D opacity TF

for c_{m-1} through c_0, or r_{m-2} through r_0, in a stepwise fashion, as shown in Fig. 3b. Note that in this figure, c_i and r_i are expressed commonly as f_i.

In our previously proposed method [12], we determined the actual opacity values empirically, because the opacity of the object projected on the screen was affected by the rendering conditions, such as the thickness of the emphasized subvolumes in the ray direction and the size of a target dataset.

In order to emphasize the internal structure of the volume dataset more clearly, it is necessary to set the opacity value higher for the area closer to the center of the dataset. In our previously proposed method [12], we assumed that the isosurfaces whose field value is higher, exist inside the dataset; thus, a higher opacity value is assigned to them, without any consideration of the thickness of the emphasized subvolumes in the ray direction. If the sampling distance is larger than the thickness, it will not be visualized regardless of the height of the corresponding opacity value. However, if the sampling distance is small, many sampling points are generated in unnecessary regions, which may cause a computational burden. To address this problem, we calculate the thickness of the emphasized region in the ray direction in a pseudo manner, and we introduce the following variable G_i into the opacity calculation formula:

$$G_i = \frac{1.0 - V_{0,i}^{1/3}}{d},\qquad(1)$$

where $V_{i,i+1}$ denotes the volume ratio of the subvolume belonging to the corresponding interval volume at $[c_{i+1}, c_i]$, and d the average of the number of grid points in the x-, y-, and z-axis directions of the target dataset. Note that $V_{i,i+1}$ and $v_{i,i+1}$ are different when multiple links of the VST are included in the field interval $[c_{i+1}, c_i]$. For example, as Fig. 2 shows, $V_{1,2}$ is calculated as the sum of the subvolume $v_{1,2}$ corresponding to link $l_{1,2}$ and a part of the subvolume $v_{0,2}$ whose scalar field is included in the interval $[c_2, c_1]$. Assuming that the homeomorphic regions are spherically distributed, $V_{0,i}$ is proportional to the radius of the sphere. Moreover, because $\Sigma_{i=0}^{m-2} V_{i,i+1}$ equals 1, $1 - V_{0,i}^{1/3}$ can be thought of as a pseudo distance from the data boundary to the corresponding region. This is further divided by d, and it is used as the thickness per grid interval.

Herein, we define the opacity value α at three control points, such as the top and two bottoms of the hat function, using the following equations:

$$\alpha(f_i) = \frac{m-1-i}{m-1}\mu G_i, \quad \alpha(f_i+\delta_i^+) = \frac{m-1-i}{m-1}G_{i-1}, \quad \alpha(f_i-\delta_i^-) = \frac{m-1-i}{m-1}G_i,\qquad(2)$$

where μ denotes a coefficient for the degree of emphasis. From this setting, the opacity values decrease as the number of voxels corresponding to the subvolume increases.

The opacity value of a scalar field value s other than the control points is calculated using Eqs. (3):

$$\alpha(s) = \begin{cases} \dfrac{m-1-i}{m-1}G_i & (f_{i+1} + \delta^+_{i+1} \le s < f_i - \delta^-_i) \\[2ex] \dfrac{(s-f_i+\delta^-_i)\alpha(f_i) - (s-f_i)\alpha(f_i-\delta^-_i)}{\delta^-_i} & (f_i - \delta^-_i \le s < f_i) \\[2ex] \dfrac{(s-f_i)\alpha(f_i+\delta^+_i) - (s-f_i-\delta^+_i)\alpha(f_i)}{\delta^+_i} & (f_i \le s < f_i + \delta^+_i) \end{cases} \qquad (3)$$

If the field interval $[-\delta^-_i, \delta^+_i]$ to be emphasized is too narrow, the sampling point of volume rendering may not appear in the region whose scalar values belong to the interval. Therefore, to secure a certain number of voxels there, our transfer function controls the width of the hat functions, as follows:

- For the critical field values ($f_i = c_i$),

$$\delta^-_i = \eta \frac{c_i - c_{i+1}}{V_{i,i+1}}, \qquad \delta^+_i = \eta \frac{c_{i-1} - c_i}{V_{i-1,i}}. \qquad (4)$$

- For the representative field values ($f_i = r_i$),

$$\delta^+_i = \delta^-_i = \eta \frac{r_i - r_{i+1}}{V_{i,i+1}}, \qquad (5)$$

where η denotes another coefficient representing the degree of emphasis. To avoid an overlap between different hat functions, for the critical field values, if $\delta^-_i > |c_i - r_i|$ then let $\delta^-_i = |c_i - r_i|$, and if $\delta^+_i > |c_i - r_{i-1}|$ then let $\delta^+_i = |c_i - r_{i-1}|$. For the representative field values, if $\delta^-_i > |r_i - c_{i+1}|$ then let $\delta^-_i = |r_i - c_{i+1}|$, and if $\delta^+_i > |r_i - c_i|$ then let $\delta^+_i = |r_i - c_i|$.

3.3 Multi-Dimensional Opacity Transfer Function

Generally, a one-dimensional transfer function is used to assign an opacity value to a field value. However, if isosurface components with the same field value have different meanings, it is impossible to obtain a visualization result that emphasizes them separately. To address this problem, we proposed a design method for multi-dimensional transfer functions based on topological attributes [13].

In this paper, we extend a two-dimensional opacity transfer function that depends on the scalar field value and the inclusion level [13]. The inclusion level is assigned for the isosurface component that has the same field value, with larger values for a more inner component. The inclusion level of the outermost isosurface component is set to 0.

Figure 3c shows an overview of our design of a two-dimensional transfer function. When the inclusion level u is even, because the isosurface component becomes outer as the scalar field value decreases, the opacity values $\alpha_m(s, u)$ are defined as $(u + 1)\alpha(s)/u_{max}$ using a one-dimensional opacity transfer function, where u_{max} denotes the maximum inclusion level. When the inclusion level u is odd, because the isosurface component becomes inner as the scalar field value decreases, opacity values $\alpha_m(s, u)$ are defined as follows:

$$\alpha_m(s, u) = \begin{cases} \dfrac{u + 1}{u_{max}} \dfrac{i}{m - 1} G_i & (f_{i+1} + \delta_{i+1}^+ < s \leq f_i - \delta_i^-) \\[2ex] \dfrac{(s - f_i + \delta_i^-)\alpha_m(f_i, u) - (s - f_i)\alpha_m(f_i - \delta_i^-, u)}{\delta_i^-} & (f_i - \delta_i^- < s < f_i) \\[2ex] \dfrac{u + 1}{u_{max}} \dfrac{i}{m - 1} \mu G_i & (s = f_i) \\[2ex] \dfrac{(s - f_i)\alpha_m(f_i + \delta_i^+, u) - (s - f_i - \delta_i^+)\alpha_m(f_i, u)}{\delta_i^+} & (f_i < s < f_i + \delta_i^+) \\[2ex] \dfrac{u + 1}{u_{max}} \dfrac{i}{m - 1} G_{i-1} & (s = f_i + \delta_i^+) \end{cases} . \quad (6)$$

4 Empirical Evaluation

From the aspects of the size of the dataset, the parameter values of a transfer function, and the LoD of the VST, we assessed their impact on the visualization results. All datasets were found to be affinely mapped to the range [0, 255]. The platform used for our experiments was a standard PC (OS: Redhat ES 7.2; CPU: Intel Xeon; Clock: 2.50 Hz; RAM: 64 GB; GPU: NVIDIA Quadro K6000).

4.1 Sensitivity to μ and η

To obtain appropriate visualization results, we can control μ and η in our transfer function design. From the definition shown in Eqs. (2), (4), and (5), μ and η affect the opacity of the accentuated isosurfaces and the width of the accentuated field interval. First, we investigated the effect of μ and η, which are the parameter values of our transfer function.

Figure 4 shows the visualization results when our method was applied to a tooth dataset ($161 \times 161 \times 161$) [15]. We used a blue-white-yellow colormap. As seen in these images, opacity increases as a μ and η increase. However, it should also be noted that the accentuated isosurfaces are not significantly emphasized when only the μ value is changed. Because a region of the accentuated field interval is narrow, it is difficult to emphasize the target isosurfaces simply by increasing the height
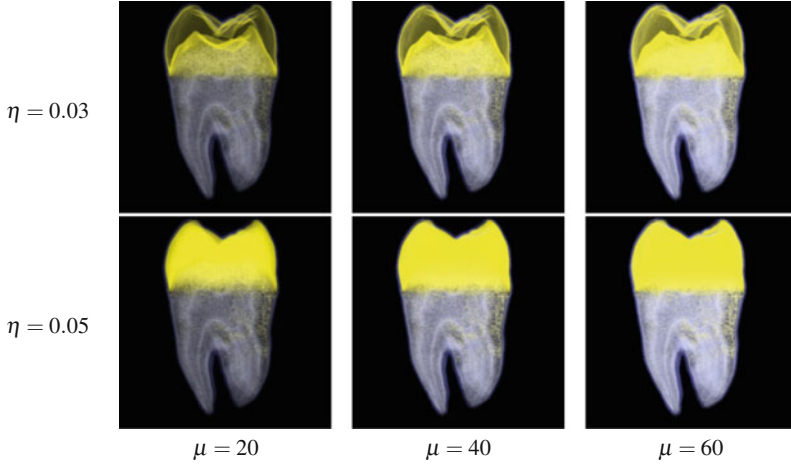
Fig. 4 Visualizing the tooth dataset with different values of μ and η, which affect the degree of emphasis of the accentuated isosurfaces and the width of the accentuated field intervals, respectively. These images show that the accentuated isosurfaces are not significantly emphasized when only the μ value is changed
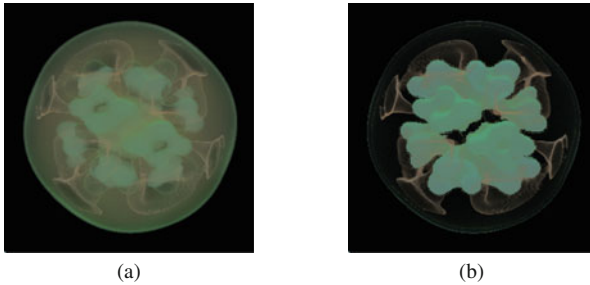

Fig. 5 Visualizing the implosion dataset with our one-dimensional transfer function and two-dimensional transfer function. These images shows our two-dimensional transfer function makes it possible to clearly visualize the inner structures by removing unnecessary surface. (**a**) One-dimensional TF. (**b**) Two-dimensional TF

of the hat function of the opacity transfer function. Therefore, an η value becomes important for this type of dataset.

Next, in order to evaluate our two-dimensional transfer function, we applied it to a laser fusion implosion dataset ($225 \times 225 \times 225$) [16], where small bubble-spike structures evolve around a contact surface between a fuel ball and pusher during the stagnation phase. The contact surface is occluded by the other outer component residing in the pusher domain, which is nothing but a phantom surface created by the action-reaction effect.

Figure 5 shows the visualization results ($\mu = 20$, $\eta = 0.05$) of the laser fusion implosion dataset, which emphasizes the representative field values with a rainbow

colormap. Figure 5a does not clearly show the inner isosurface components of interest, because these are indeed occluded by the outer phantom surface. On the other hand, Fig. 5b clearly illustrates that our two-dimensional transfer function makes it possible to clearly visualize the inner structures by removing unnecessary phantom surface.

4.2   Dataset Size Sensitivity

Next, we evaluated whether similar visualization results can be obtained from the datasets with different sizes. In order to demonstrate the effectiveness of our method, we applied it to two types of volume datasets.

The first is an analytical dataset [12], which is formulated as follows:

$$
f(x, y, z) = 4c^2 \left((x - R)^2 + (z - R)^2\right) - \left((x - R)^2 + y^2 + (z - R)^2 + c^2 - d^2\right)^2
$$
$$
+ 4c^2 \left((x + R)^2 + (z + R)^2\right) - \left((x + R)^2 + y^2 + (z + R)^2 + c^2 - d^2\right)^2, \quad (7)
$$

where $0 < d < c$ and $c^2 + d^2 \geq 6R^2$. This dataset has two maxima, three saddles, and a virtual minimum. Figure 6 shows the VST of the analytical function and the visualization results applied to three analytical datasets with different sizes: $33 \times 33 \times 33$, $65 \times 65 \times 65$, and $129 \times 129 \times 129$. As Fig. 6 shows, our transfer functions to accentuate the critical field values with a blue-white-red colormap ($\mu = 30$, $\eta = 0.035$). The volume ratio corresponding to each link differs according to the dataset size, as shown in Table 1. It is clear that the volume ratios are nearly equal regardless of the size of dataset. The results presented in Figs. 6b–d show that the proposed transfer functions provide similar visualization results regardless of the size of the dataset.

The second is a three-dimensional head dataset, which was provided courtesy of Siemens Medical Systems, Inc. (Iselin, NJ). Figure 7 shows the visualization results applied to two head datasets with different sizes: $65 \times 65 \times 65$ and $129 \times 129 \times 129$. These images use our transfer functions to accentuate the critical field values with a blue-white-yellow colormap ($\mu = 40$, $\eta = 0.05$). The figure illustrates that the visualization results obtained for a real-world dataset are similar to those obtained for the analytical dataset, regardless of the size of the dataset.

These results demonstrate that the effectiveness of our method does not depend on the size of datasets, even for different types of datasets.
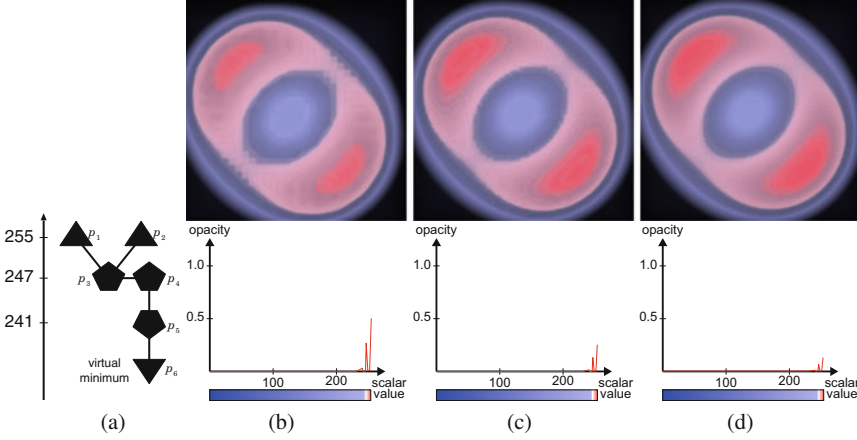
Fig. 6 VST and visualization results for the analytical datasets with our opacity transfer functions ($\mu = 30$, $\eta = 0.035$). These images show that our transfer functions provide similar visualization results regardless of the size of the dataset. (**a**) VST. (**b**) $33 \times 33 \times 33$. (**c**) $65 \times 65 \times 65$. (**d**) $129 \times 129 \times 129$

Table 1 Volume ratios corresponding to each link of the analytical datasets shown in Fig. 6

Link	$33 \times 33 \times 33$	$65 \times 65 \times 65$	$129 \times 129 \times 129$
l_{1-3}	4.67e−2	4.86e−2	4.96e−2
l_{2-3}	4.67e−2	4.86e−2	4.96e−2
l_{3-4}	0.0	0.0	0.0
l_{4-5}	1.73e−1	1.82e−1	1.86e−1
l_{5-6}	7.33e−1	7.21e−1	7.15e−1

4.3 Response to the LoD of the VST

Finally, in order to evaluate whether our method is effective even if the number of the accentuated surfaces increases, we examined how our transfer function responds to differences in the LoD of the VST. When the VST extracts the topological structure from the target dataset, it may contain a large number of minor critical points. Because they may hide the important global structure of the dataset, we need to simplify the extracted VST by removing them.

We applied our method to a nucleon dataset ($41 \times 41 \times 41$) [17] to evaluate the effect of the LoD of the VST. Figure 8 shows the visualization results with a heat object colormap ($\mu = 20$, $\eta = 0.04$); the results emphasize the representative field values for different LoDs of the VST. Figure 8a shows a visualization result based on the original VST, which has 1006 critical points and 21 critical field values. Note that we cannot show the original VST because it is too complicated to draw. Figures 8b, c show the results based on simplified VSTs that have ten and six critical
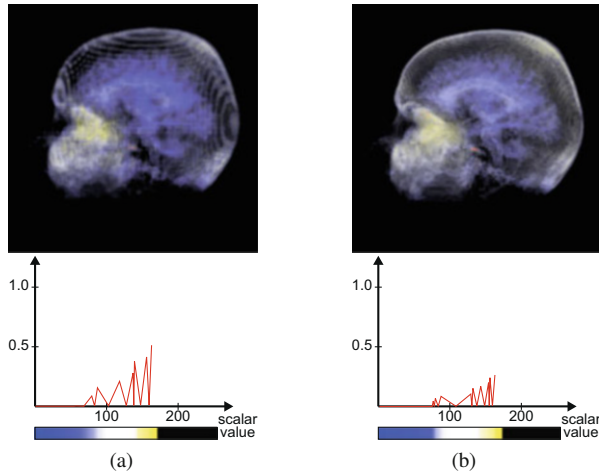
Fig. 7 Visualization results of the head datasets with our opacity transfer functions ($\mu = 40$, $\eta = 0.05$). These images show that our transfer functions provide similar visualization results regardless of the size of the dataset. (**a**) $65 \times 65 \times 65$. (**b**) $129 \times 129 \times 129$
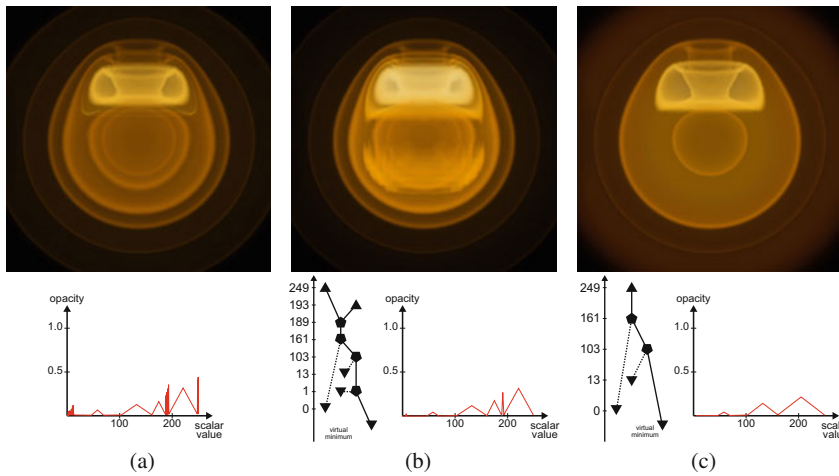

Fig. 8 Visualizing the nucleon dataset applied for different LoDs of the VST. These images show that our transfer function can effectively visualize the topological structure even if the number of critical points decreases. (**a**) Original VST (1006 critical points). (**b**) Simplified VST (10 critical points). (**c**) Simplified VST (6 critical points)

points and six and five critical field values, respectively. These images illustrate that our transfer function can effectively visualize the topological structure, regardless of the number of critical points.

5   Further Controllability

In this section, we evaluate our method further from the perspective of image quality and processing speed.

5.1   Image Quality

In order to evaluate the quality of a visualization result, we defined an evaluation function based on a normalized Shannon information entropy as follows:

$$Entropy = \frac{-\sum_{i=0}^{M} p(A_i) \log_2 p(A_i)}{\log_2 M}, \tag{8}$$

where M and $p(A_i)$ represent the number of levels and the probability of pixels of level i, respectively. When the value of entropy is high, there are many pixels at various levels, which means that the information entropy of the visualized image is large. Conversely, when there are numerous pixels with the same levels, the information entropy is small. Therefore, we assume that the higher the entropy, the better the quality of the image. Note that the evaluation function can also potentially lead to misleading results in the presence of noise in the dataset. We evaluate the quality of images in terms of their hue, saturation, and luminance values. When the entropies of hue, saturation, and luminance are represented by E_h, E_s, and E_l, respectively, our evaluation function is defined as follows:

$$\begin{cases} E = k_H E_H + k_S E_S + k_L E_L \\ k_H + k_S + k_L = 1 \qquad (0 \le k_H, k_S, k_L \le 1) \end{cases}, \tag{9}$$

where k_H, k_S, and k_L are the weight coefficients for hue, saturation, and luminance, respectively. Users can control these coefficients based on their preferences.

Figure 9a shows the heatmap that represents our evaluation function, where $k_H = k_S = k_L = 1/3$ of the analytical dataset ($129 \times 129 \times 129$). The value for μ was increased by 10 increments ranging from 10 to 50. The value for η was increased by 0.005 increments, and its range was set so that the field interval that emphasizes the critical field value does not exceed the adjacent representative field values. The cells surrounded by the black box and the white box represent the minimum value and the maximum value of the corresponding entropy, respectively. Figures 9b, c show the best and worst visualization results. The values of μ and η in the best case are 30 and 0.035, respectively; in the worst case, they are 10 and 0.005. Moreover, in the best case, the value of our evaluation function is 0.88; in the worst case, it is 0.79. In this example, the images in the worst cases are darker than the images representing the best cases. By introducing our evaluation mechanism, we can automatically obtain an appropriate visualization result, which is well balanced among hue, saturation,
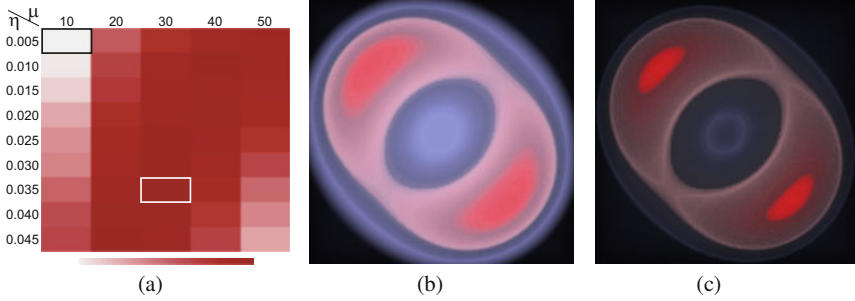
Fig. 9 Entropy heatmap of E and the best and worst visualization results of the analytical dataset. We can automatically obtain an appropriate visualization result referring to the entropy heatmap. (**a**) Entropy heatmap of E ($k_H = k_S = k_L = 1/3$). (**b**) Best ($\mu = 30, \eta = 0.035$). (**c**) Worst ($\mu = 10, \eta = 0.005$)

and luminance, from the entropy heatmap, as shown in Fig. 9a. Note that μ and η in the best case were used to obtain Fig. 6.

5.2 Processing Speed

In our method, the time required for the transfer function design, excluding the VST extraction, was 0.01 CPU seconds for any dataset throughout this paper. However, the computation time required for VST extraction also increases as the dataset size increases. Our extraction algorithm [12, 18] can adjust the accuracy of the VST extraction with interactive operations, although it may take several minutes to several tens of minutes of processing time in some cases. Though a discussion of the extraction algorithm of the VST is beyond the scope of this paper, we had to accelerate our algorithm to extract the topological structures from large-scale datasets.

In order to shorten the processing time required for the VST extraction, it is conceivable to use downsized datasets to obtain the VST and to reduce the processing time. To visualize a large-scale dataset, we design the transfer function based on the VST extracted from the downsized dataset and then visualize the original large dataset using that function. The experiments presented in Sect. 4.2 that similar visualization results can be obtained even if the VST extracted from the downsized dataset is used.

Figure 10 shows the VST and the visualization result of the stag beetle dataset ($832 \times 832 \times 494$) [19], which emphasizes the critical isosurfaces. The VST was extracted from the small stag beetle dataset ($208 \times 208 \times 123$) and simplified until the number of the critical points became 14. As the image illustrates we can obtain an effective visualization result even if the transfer function is designed based on the VST extracted from the downsized dataset.
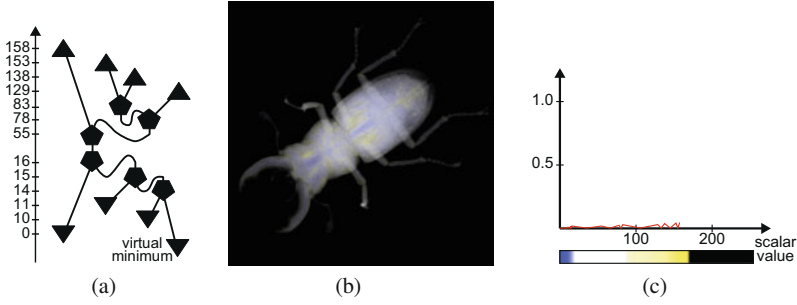
Fig. 10 VST, visualization result, and transfer function of the stag beetle dataset. We can obtain effective visualization results even if the transfer function is designed based on the VST extracted from the downsized dataset. (**a**) VST. (**b**) Visualization result. (**c**) Transfer function

6 Conclusion

In this paper, we proposed a method to define the opacity values of topology-accentuated transfer functions. Controlling the height and width of the hat functions made it possible to emphasize the feature isosurfaces regardless of the kind of dataset. According to our empirical evaluations so far, μ and η should range from 10 to 50 and from 0.01 to 0.05, respectively, because the internal structure becomes invisible outside of the range in many cases. Our multi-dimensional transfer function is also able to more clearly emphasize the inner structures in a target dataset. In addition, our empirical evaluations suggest that the proposed transfer function design can automatically provide guaranteed results regardless of the size of the target dataset and the LoD of the VST. Furthermore, by introducing the evaluation function with different combinations of color components, we can anticipate appropriate visualization results based on users' preferences.

The present transfer function design can incorporate sampling distance as a variable of its definition in order to determine the region to be emphasized around accentuated isosurfaces. In order to evaluate the effects of the designed transfer functions in more detail, we will apply them to more complicated and larger datasets in future studies. Moreover, we should take up the challenge of considering parameter settings for a multi-dimensional transfer function based on multi-variate topological structures [20].

References

1. Ljung, P., Krüger, J., Gröller, E., Hadwiger, M., Hansen, C.D., Ynnerman, A.: State of the art in transfer functions for direct volume rendering. CGF **35**(3), 669–691 (2016)
2. Kindlmann, G., Durkin, J.W.: Semi-automatic generation of transfer functions for direct volume rendering. In: Proceedings of the IEEE Symposium on Volume Visualization, pp. 79–86 (1998)
3. Kniss, J., Kindlmann, G., Hansen, C.: Multidimensional transfer functions for interactive volume rendering. IEEE TVCG **8**(3), 270–285 (2002)
4. Hladůvka, J., König, A., Gröller, E.: Curvature-based transfer functions for direct volume rendering. In: Proceedings of the Spring Conference on Computer Graphics 2000, pp. 58–65 (2000)
5. Kindlmann, G., Whitaker, R., Tasdizen, T., Möller, T.: Curvature-based transfer function for direct volume rendering: methods and applications. In: Proceedings of the IEEE Vis 2003, pp. 513–520 (2003)
6. Weber, G., Dillard, S., Carr, H., Pascucci, V., Hamann, B.: Topology-controlled volume rendering. IEEE TVCG **13**(2), 330–341 (2007)
7. Zhou, J., Takatsuka, M.: Automatic transfer function generation using contour tree controlled residue flow model and color harmonics. IEEE TVCG **15**, 1481–1488 (2009)
8. Carr, H., Snoeyink, J., Panne, M.V.D.: Flexible isosurfaces: simplifying and displaying scalar topology using the contour tree. Comput. Geom. Theory Appl. **43**(1), 42–58 (2010)
9. Praßni, J.-S., Ropinski, T., Mensmann, J., Hinrichs, K.H.: Shape-based transfer functions for volume visualization. In: Proceedings of the IEEE PacificVis 2010, pp. 9–16 (2010)
10. Xiang, D., Tian, J., Yang, F., Yang, Q., Zhang, X., Li, Q., Liu, X.: Skeleton cuts—an efficient segmentation method for volume rendering. IEEE TVCG **17**(9), 1295–1306 (2011)
11. Fernandes, O., Frey, S., Ertl, T.: Interpolation-based extraction of representative isosurfaces. In: Advances in Visual Computing, pp. 403–413. Springer International Publishing, Cham (2016)
12. Takahashi, S., Takeshima, Y., Fujishiro, I.: Topological volume skeletonization and its application to transfer function design. Graph. Models **66**(1), 24–49 (2004)
13. Takeshima, Y., Takahashi, S., Fujishiro, I., Nielson, G.M.: Introducing topological attributes for objective-based visualization of simulated datasets. In: Proceedings of the International Workshop on Volume Graphics, pp. 137–236 (2005)
14. Fujishiro, I., Maeda, Y., Sato, H., Takeshima, Y.: Volumetric data exploration using interval volume. IEEE TVCG **2**(2), 144–155 (1996)
15. Pfister, H., Lorensen, B., Bajaj, C., Kindlmann, G., Schroeder, W., Avila, L.S., Martin, K., Machiraju, R., Lee, J.: The transfer function bake-off. IEEE Comput. Graph. Appl. **21**(3), 16–22 (2001)
16. Sakagami, H., Murai, H., Seo, Y., Yokokawa, M.: 14.9 TFLOPS three-dimensional fluid simulation for fusion science with HPF on the Earth Simulator. In: Proceedings of the Supercomputing 2002 (2002)
17. Sonderforschungsbereiche 382 of the German Research Council. Nucleon.
18. Takahashi, S., Nielson, G.M., Takeshima, Y., Fujishiro, I.: Topological volume skeletonization using adaptive tetrahedralization. In: Proceedings of the GMP 2004, pp. 227–236 (2004)
19. Gröller, M.E., Glaeser, G., Kastner, J.: Stag beetle (2005). https://www.cg.tuwien.ac.at/research/publications/2005/dataset-stagbeetle/.
20. Carr, H., Duke, D.: Joint contour nets. IEEE TVCG **20**(8), 1077–2626 (2014)