Surface carving-based automatic volume data reduction

Qichao Wang, Yubo Tao & Hai Lin

The Visual Computer
International Journal of Computer Graphics

ISSN 0178-2789
Volume 31
Number 11

Vis Comput (2015) 31:1459-1470
DOI 10.1007/s00371-014-1026-2

Springer

Vis Comput (2015) 31:1459–1470
DOI 10.1007/s00371-014-1026-2

ORIGINAL ARTICLE

Surface carving-based automatic volume data reduction

Qichao Wang · Yubo Tao · Hai Lin

Published online: 21 September 2014

Abstract Many fields, such as medicine and biology, are producing an increasingly large volume using high-resolution digital imaging techniques, and this makes effective data analysis and visualization of these volumes more and more difficult. Volume reduction, decreasing the volume size, is one of the promising directions to solve this challenge for interactive volume visualization. In this paper, we present an automatic volume data reduction method called surface carving. It intelligently removes contextual voxels while preserving important features, and finally generates an optimal volume at the desired reduction size/rate. For large volume data sets, a multilevel banded method is introduced to gracefully overcome the memory limit and speed up volume reduction. We compare our technique with traditional cropping and scaling approaches and demonstrate the effectiveness and efficiency of our method with several volume data sets.

Keywords Automatic · Volume reduction · Surface carving

1 Introduction

Volume visualization has been widely used to explore 3D volume data sets in different fields, ranging from medicine to geophysics. With the development of various high-resolution digital imaging techniques, the size of the acquired volume is increased significantly. These large volumes make the data analysis and visualization tasks on desktop machines increasingly difficult, even impossible for mobile devices due to the limited resources. Volume reduction is a common solution to overcome this challenge for interactive volume visualization applications, such as mobile applications with limited screen size [1] and web applications with limited internet bandwidth [2].

The most popular volume reduction methods are volume downsampling and cropping. Standard volume downsampling methods are often used in the hierarchical multi-resolution volume construction, such as subsampling [3] and kernel-based filtering [4]. However, they generally treat the data feature and context, i.e., unimportant features/voxels, uniformly, which results in unrecognizable tiny features and artifacts. Cropping could remove the context, but it only excludes voxels from the volume periphery due to the box geometric constraint and may also remove the feature itself. Although the cropping method could be much easier to control the reduction rate than the downsampling method, it is not flexible and effective enough to select the content to reduce. A more effective volume reduction method should intelligently remove the contextual voxels with a more flexible geometric constraint while preserving important features.

In this paper, we propose an automatic volume reduction method called surface carving. It is a multi-pass method to iteratively and successively reduce the volume until it reaches the desired size or reduction rate. In each pass, surface carving searches for a connected manifold surface with the least distortion or artifact in the volume space among the x, y, and z directions, and carves out the voxels on the connected manifold surface to achieve volume reduction. In each direction, finding the carved surface is formulated as a minimum energy problem, which first constructs a volume graph based

Q. Wang · Y. Tao (✉) · H. Lin
State Key Lab of CAD&CG, Zhejiang University, Hangzhou, People's Republic of China
e-mail: taoyubo@cad.zju.edu.cn

Q. Wang
e-mail: wqcriver@gmail.com

H. Lin
e-mail: lin@cad.zju.edu.cn

on the saliency value of each voxel, then applies the minimum cost graph cut algorithm to generate a connected and less important manifold surface. To remove contextual voxels while preserving important features in the reduced volume, the voxel's saliency value can be derived from the volume itself, such as the gradient and curvature, and be specified by the user through the transfer function. As the memory requirement of the volume graph is increasing with the size of the volume, a multilevel banded method is introduced to overcome the memory limit in reducing large volume data sets, and it also improves the computational efficiency of surface carving.

The main contributions of this paper are:

(1) proposing surface carving to implement volume data reduction;
(2) designing a novel volume saliency measure to drive volume reduction;
(3) introducing a multilevel banded approach for large volumes to overcome the memory limit and accelerate the reduction process;

The paper is structured as follows. The related work is discussed in Sect. 2. Section 3 describes the automatic volume data reduction in detail. We show several examples of surface carving and discuss the large data reduction and progressive transmission and reconstruction applications in Sect. 4.

2 Related work

As volume reduction can be considered as one volume editing operation, we first review previous volume editing operations. Direct volume editing, proposed by Bürger et al. [5], introduces 3D spherical brushes to interactively edit the volume data on GPUs, for example user-guided feature coloring, erasing, segmentation and annotation. Yuan et al. [6] presented two-pass graph cut algorithm to cut out 3D volumetric features based on user's strokes in the rendered image. Zhao et al. [7] reported an intelligent volume brush, iVolBrush, for effective 3D painting on the volume. WYSIWYP (What You See Is What You Picking) [8] allowed the user to intuitively select the most visible semi-transparent feature displayed in the rendered image accurately.

Giachetti et al. [9] presented an edge-directed volume supersampling method for high-quality rendering of large medical data sets. This method keeps constant the original energy of the subdivided voxel and optimizes edge continuity in the volume upscaling process. Wang et al. [10] proposed a volume upscaling method based on local self-examples to make feature analysis more accurate and efficient. Our method is an inverse operation to volume upscaling, reducing the size for interactive visualization of large volumes.

Volume downsampling and cropping are two common volume reduction approaches for large volume visualization. Volume downsampling has been used in multi-resolution data representation for rendering large volume data sets interactively [3,11]. It usually applies uniform subsampling, or kernel-based filtering techniques to the volume and constructs a multi-resolution data hierarchy. However, this homogeneous downsampling does not distinguish feature and context and would blur the important features in the low-resolution volume. Volume cropping also has the functionality of volume reduction. In the crop-and-zoom method [12], the user defines a volume-of-interest using a bounding box, to crop and zoom the sub-volume out of the large volume. Although cropping removes unimportant voxels in the periphery of the interested region, the sub-volume may contain a lot of context or exclude part of prominent features close the periphery. The proposed method intelligently removes contextual voxels with a flexible geometric constraint and keeps important features.

Image resizing is an active research topic in computer graphics. Avidan and Shamir [13] presented seam carving for content-aware image resizing. It searches for a connected path of less important pixels by dynamic programming. This idea has been extended to video retargeting by Rubinstein et al. [14]. As dynamic programming cannot be directly used for video retargeting, they derived an equivalent implementation by properly constructing the graph and resizing the frames by the graph cut algorithm. The proposed surface carving also generates a manifold surface with the least energy by the graph cut algorithm, and carves out this surface to achieve volume reduction while preserving salient features.

The idea of the proposed volume data reduction is also very similar to the focus + context technique in volume visualization, preserving important features while removing or suppressing less important contextual features. Viola et al. [15] presented importance-driven volume rendering to highlight important features by assigning them a high importance. Wang et al. [16] introduced an energy optimization model for the focus + context visualization. It deforms the volume space based on the importance value of each voxel to magnify features of interest. The deformed volume space can also be resampled to generate a reduced data set with better preserved features. Recently, the conformal magnifier proposed by Zhao et al. [17] also magnifies a region of interest by conformal mapping for the volume. It enlarges the user-specified region of interest and suppresses the context without any cropping. Our method is also based on the saliency value of each voxel/feature, which could be defined by the user like the importance value and user-specified features in the previous focus + context researches. The volume reduction process can also be considered a deformation process, deforming the less important and contextual surfaces while magnifying important features implicitly. Unlike the continuous manner of Wang et al. [16], which may not heavily shrink the unimportant regions, our method can discretely handle removal of the unnecessary content well.

3 Automatic volume reduction based on surface carving

The proposed multi-pass volume reduction method, surface carving, intelligently removes less important or contextual voxels while preserving salient features, and finally generates a reduced volume at the desired size or reduction rate. In each pass, surface carving reduces the chosen dimension by one by finding and removing an unimportant or contextual manifold surface in the other two dimensions based on voxels' saliency values. For example, surface carving can reduce the volume with the size $M \times N \times K$ to $M \times N \times (K-1)$ in one pass, and the carved surface is made of $M \times N$ voxels. This surface must satisfy with two properties: monotonicity and connectivity. Monotonicity means that the surface must include one and only one voxel in each row in the reduced dimension, i.e., the surface should not overlap in the reduced dimension to keep the shape of the reduced volume as a box. Connectivity requires that the voxels of the surface must be connected to minimize visual artifacts resulting from the voxel removal.

To satisfy these requirements, surface carving is derived from seam carving for image resizing [13,14]. Dynamic programming and graph cuts are two general computational methods to find an optimal seam for image resizing. As dynamic programming can not be directly extended to 3D, this paper employs graph cuts to search for the optimal surface. Our pipeline is illustrated in Fig. 1. Users can specify the reduction size/rate. In each iteration for the input volume, a volume graph is first constructed with each voxel as a node. The connection between nodes is based on the backward scheme [14], and the energy costs of edges are defined from voxels' saliency values. These values can be computed from the volume data set itself, such as the gradient magnitude, or specified by the user through the transfer function. Then, surface carving is formulated as a minimum cost graph cut problem. We use the graph cut algorithm to find the optimal surface. Finally, the voxels of the optimal surface are carved and the remaining voxels are packed into a reduced volume for further processing. After several iterations of surface carving, we can reduce the original volume data set to a new volume with the specified size or reduction rate.

We will first describe the method for finding an unimportant or contextual manifold surface for a given dimension, and then show how to use this method to achieve automatic data reduction with the specified size or reduction rate.

3.1 Volume graph construction

A volume graph $G = \{V, E\}$ is constructed for the input volume. As each voxel is a node in the volume graph, the node set $V = \{v_1, v_2, \ldots, v_n, S, T\}$ contains n nodes representing n voxels, and two virtual terminal nodes S (Source) and T (Sink).

The edge set E connects the nodes based on the neighborhood relationships among voxels and the selected dimension. The directed edge with the direction from the source S to the sink T is called the forward edge, otherwise it is the backward edge. Let the selected dimension to be reduced is the Z direction shown in Fig. 2. The terminal node S is connected with forward infinite cost edges to all voxels on the boundary $X-Y$ plane with the minimum Z value (the leftmost plane of the volume shown in Fig. 2). Similarly, the terminal node T is connected with forward infinite cost edges from all voxels on the boundary $X-Y$ plane with the maximum Z value (the rightmost plane of the volume shown in Fig. 2).

Each voxel is connected with different cost edges to its neighborhood voxels. A typical voxel $v(x, y, z)$ is connected with two neighbor voxels $v(x, y, z \pm 1)$ by forward edges with the cost C and backward infinite cost edges. The voxel $v(x, y, z)$ is also connected with eight diagonal voxels $v(x \pm 1, y, z \pm 1)$ and $v(x, y \pm 1, z \pm 1)$ by backward infinite cost edges. Figure 2 shows an example for the voxel node in blue with different edges indicated in solid black lines. In summary, all backward edges are assigned with the infinite

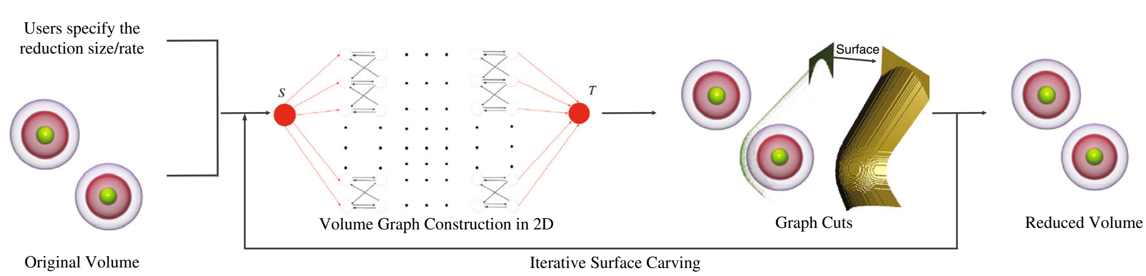


Fig. 1 The pipeline of automatic volume reduction. Users specify the reduction size/rate. The saliency value of each voxel is derived from the volume itself, or specified by the user through the transfer function. Surface carving first constructs a volume graph satisfying monotonicity and connectivity properties, and then applies the minimum cost graph cut algorithm to find a less important or contextual manifold surface. This process is iteratively applied to the reduced volume to create a target reduced volume while preserving important features

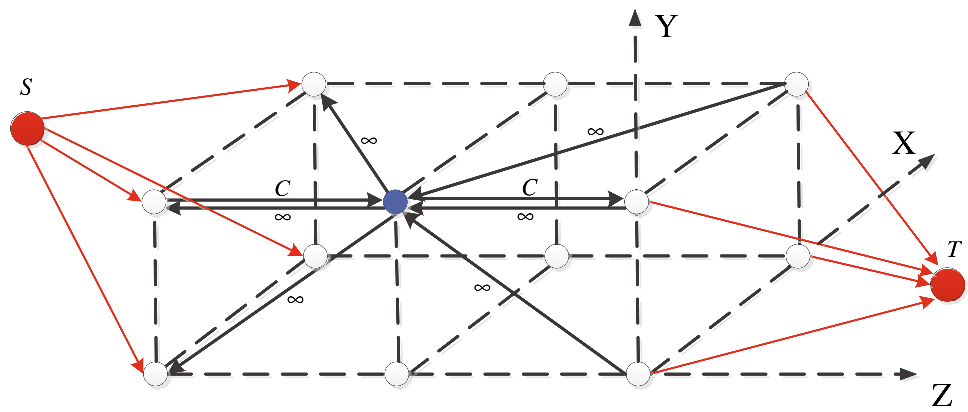


Fig. 2 The illustration of volume graph construction. The selected dimension to be reduced is the Z direction. Virtual terminal nodes in *red*, S and T, are connected with infinite weight edges to all voxels on the leftmost and rightmost $X-Y$ plane of the volume respectively. The voxel in *blue* is connected to its neighboring voxels with different edge costs in *solid black lines*. For forward directed edges (in the direction from S to T), the edge cost is C depending on the voxels of the edge, and all backward directed edges (in the direction from T to S) are with infinite weights

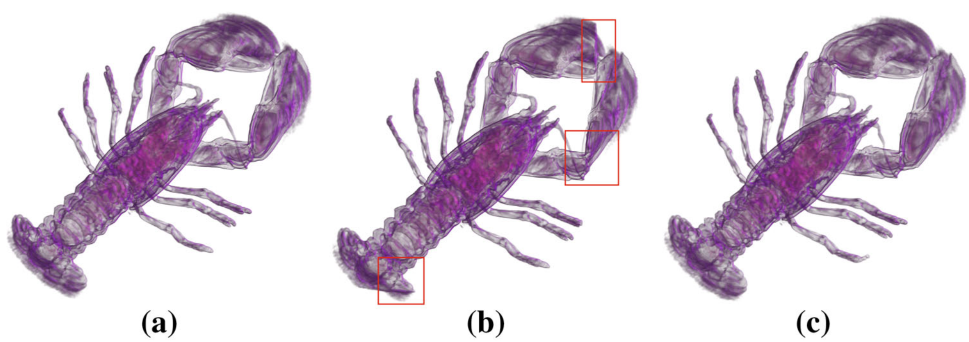


Fig. 3 Volume reduction without/with the spatial continuity constraint. The lobster volume is reduced from $254 \times 248 \times 56$ to $242 \times 236 \times 56$. **a** The original volume. The reduced volume **b** without and **c** with the spatial continuity constraint. We can see that the reduced volume shows better structure preserving if the constraint is considered

cost, while all forward edges among voxels are assigned with the cost C, which is based on saliency values of two voxels of the edge and will be discussed in the next section.

As this graph construction based on the backward scheme has been proven satisfying monotonicity and connectivity [14], the surface generated from the graph cut algorithm is monotonic and connected to preserve continuity in the reduced dimension and to avoid jittery artifacts. Similar to the temporal continuity constraint in [14], our method considers the spatial continuity constraint in the other two dimensions (the X and Y dimensions in Fig. 2). Without the spatial continuity constraint, simply applying the seam carving operator defined in [13] separately to each slice of the volume would introduce serious distortions (Fig. 3).

3.2 Edge energy computation

Since surface carving is formulated as a minimum energy problem, finding the less important or contextual surface is equivalent to calculating the surface with the least amount of energy. Thus, the definition of the edge cost C plays a crucial role in the proposed volume reduction. Different specifications of the edge cost would lead to different volume reduction results.

In surface carving, the edge cost C is based on the saliency values of voxels. As the meaning of saliency depends on the application and the user, it is more suitable to provide a generic way to include any saliency measure. Specially, we consider two general saliency measures. One is the intrinsic properties of the volume itself, such as the gradient magnitude and the curvature. The other is the user specification, such as the transfer function and the application-specific term.

The saliency value for the voxel v is a weighted combination of these two measures as follows:

$$S(v) = \lambda P(v) + (1-\lambda) Q(v), \quad (1)$$

where λ is a constant balancing the two components. $P(v)$ is the normalized intrinsic property measure, and $Q(v)$ is the user's saliency specification. A large λ makes the reduced

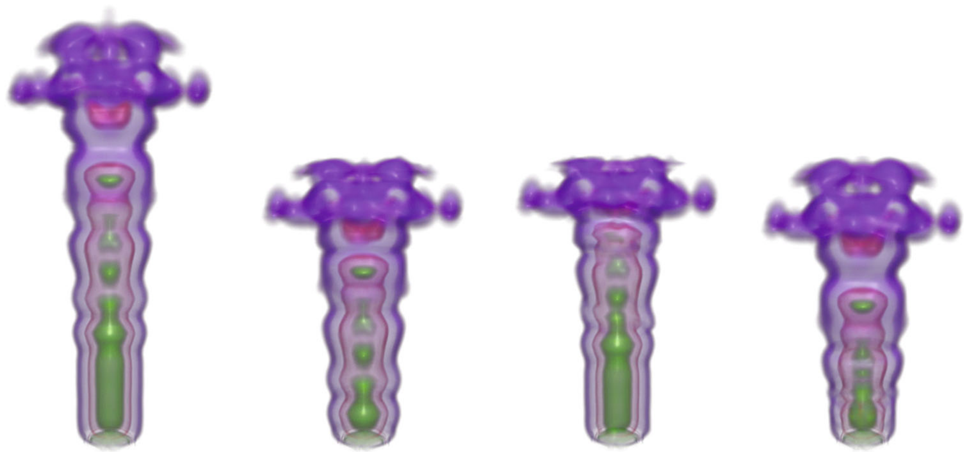


Fig. 4 Different volume reduction results based on different saliency measures for a fuel volume. The volume is reduced from 64 × 64 × 64 to 64 × 64 × 42. **a** The original volume. The reduced volume based on **b** the gradient magnitude, **c** the curvature and **d** the opacity transfer function

result keep voxels with a more important intrinsic property while a small one preserves the salient structure specified by the user. The edge cost C between the voxel $v(x, y, z)$ and $v(x, y, z + 1)$ is defined as $C(v(x, y, z), v(x, y, z + 1)) = S(v(x, y, z))$, and it is similar for other dimensions.

3.2.1 Intrinsic property measure

In volume visualization, users are usually interested in boundary regions between homogeneous materials, and the gradient magnitude is a well-defined property for the boundary regions [18]. Hence, the gradient magnitude can be considered as the saliency value for each voxel. A voxel with a high gradient magnitude may contain more salient information, while a voxel with a low gradient magnitude is less important or contextual.

The discrete gradient magnitude estimation for the voxel v is defined as follows:

$$P(v) = \|\triangledown f(v)\|, \quad (2)$$

where $f(v)$ is the scalar value of the voxel v, and $\triangledown$ is the gradient operator. When the gradient magnitude is used as the edge cost in surface carving, the boundary of features will be maintained while the voxels with low gradient magnitudes are removed.

Besides the gradient magnitude, any intrinsic property, such as the curvature and the symmetry, is a feasible saliency measure for the volume and can be integrated into this intrinsic property measure. Figure 4 compares the surface carving results for a fuel data set using different saliency measures. Figure 4b, c is the volume reduction results based on the gradient magnitude and the curvature as the saliency measure, respectively. Compared with the original volume data in Fig. 4a, the top part of the fuel volume is preserved due to its large gradient magnitude or curvature. More homogeneous regions, like the bottom part of the volume, are likely to be discarded for its low gradient magnitude in Fig. 4b. However, these regions are preserved due to its high curvature in Fig. 4c.

3.2.2 User specification measure

As the voxels with the same intrinsic property may have different importance for different applications, users' prior knowledge about important features should be also included in the saliency measure. For example, the transfer function is a widely used tool to allow users to classify features and specify importance for each feature. Important features are usually assigned with a high opacity to make them more visible, while contextual features are assigned with a low opacity to make them transparency. Thus, voxels with a higher opacity generally means more important, and we can define the opacity of the voxel as the saliency value of this voxel:

$$Q(v) = \alpha(v), \quad (3)$$

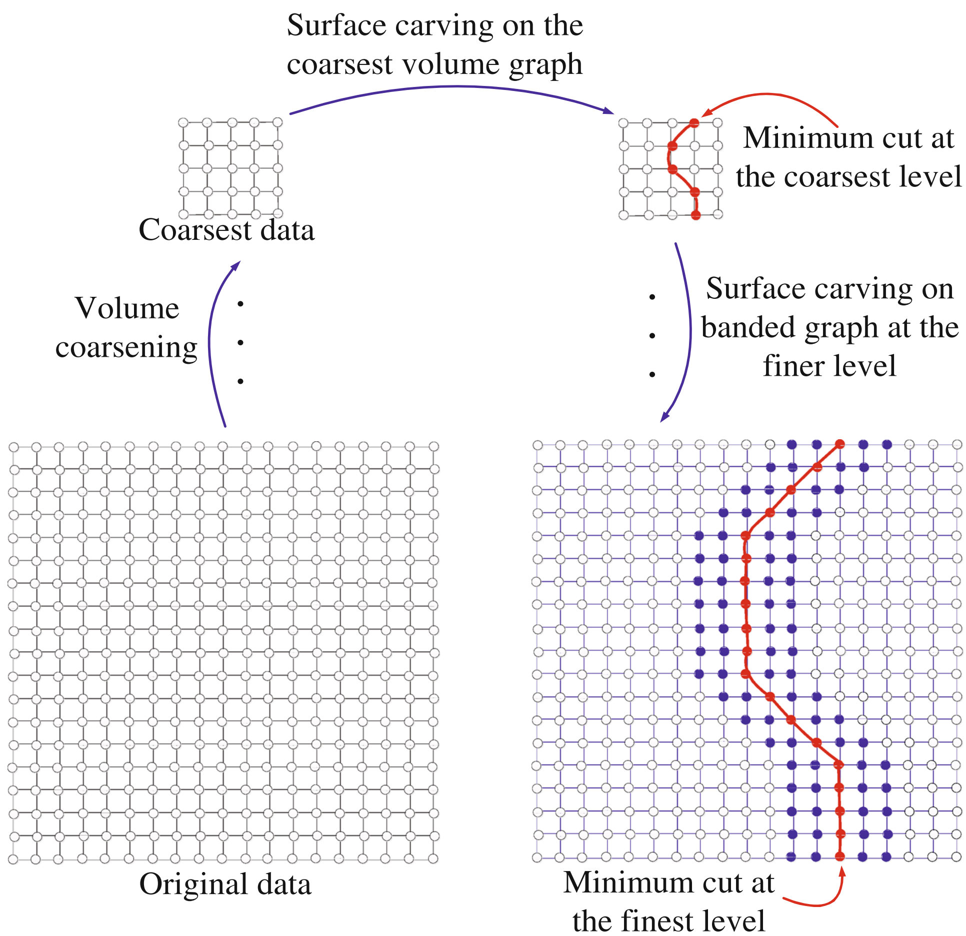
where $\alpha(v)$ is the opacity of the voxel v. Meanwhile, the color differences between neighboring voxels are the boundaries specified by the user, and can also be included in the saliency measure.

Figure 4d is the volume reduction result based on the opacity transfer function. As can be seen clearly from these figures, when the user specifies different saliency values for voxels, the proposed surface carving can automatically identify less important voxels and construct a manifold surface to discard its voxels.

3.3 Graph cuts in surface carving

After the construction of the volume graph and the specification of edge costs, the graph cut algorithm [19] is applied to find the optimal less important or contextual manifold surface. An S/T cut (or a surface cut) in the volume graph is defined as a partitioning of the nodes into two disjoint subsets

Fig. 5 The illustration of the banded multilevel method in 2D



S_{set} and T_{set} such that $S \in S_{set}$ and $T \in T_{set}$. The cost of the cut is the sum of the cost of the boundary edges across these two sets, defined as follows:

$$C_{cut}^{d} = \sum_{(v_p,v_q)\in E, v_p\in S_{set}, v_q\in T_{set}} C(v_p, v_q), \tag{4}$$

where v_p and v_q are two neighborhood nodes along the reduced dimension d. Note that a cut cost is the sum of the cost of directed forward edges. The optimal surface is obtained by minimizing the cut cost C_{cut}^{d}, and this is the minimum cost among all valid cuts. Based on the optimal surface, the voxels of the surface are carved and the remaining voxels are shifted and packed into a reduced volume for further processing.

The computational time of surface carving depends on the numbers of nodes and the number of edges in the graph, which is approximately five times the number of nodes. The memory requirement for the volume graph is also proportional to the number of voxels in the volume. For large volumes, larger than $256 \times 128 \times 128$, the memory required by the volume graph algorithm exceeds the limit allowed in a typical 32-bit personal computer. In addition, computing the minimal cut for the volume graph of such large volume is time-consuming, maybe prohibitive, due to the polynomial worst- case complexity [20].

Several acceleration methods have been proposed to improve the graph cut algorithm in the image segmentation field. Lombaert et al. [20] introduced a multilevel banded graph cut method to reduce the memory requirement for large images. Kohli and Torr [21] computed minimum cuts on an updated graph, with speed gains up to two orders of magnitude. Vineet and Narayanan [22] implemented the graph cut algorithm on the GPU to accelerate the image segmentation algorithm. As the main challenge of surface carving is the memory requirement for the large volume data set, we employ the banded multilevel method [20] and extend it to the volume graph.

The general process of the banded multilevel surface carving is illustrated in Fig. 5 using the 2D case. First, the input volume is coarsened using a standard multi-resolution technique. Then, a minimal cut is computed on the coarsest volume graph constructed from the coarsest volume, as shown the top right part of Fig. 5. The cut nodes are projected on the successive higher resolution volume, and this results in a narrow banded volume, which limits the candidate nodes to

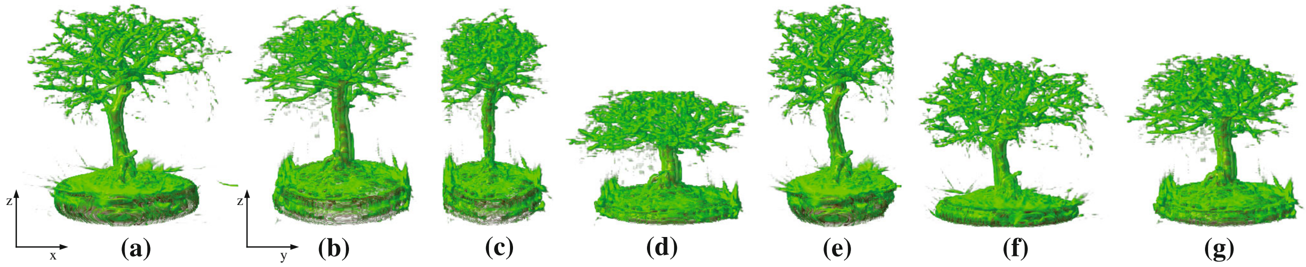


Fig. 6 Volume reduction results with the reduction rate of about 50 %. **a**–**b** The original bonsai volume from different views. **c**–**e** Each volume reduced along the x, y, z dimensions, respectively, resulting in clear visual artifacts. **f**–**g** The reduced volume using the proposed surface carving. The views of **f**, **g** are the same with these of **a**, **b**. Given the reduction rate, our method preserves important features and detail information as much as possible while removing contextual voxels

be extracted from the coarsest graph. The width of the narrow band could be controlled by the distance dst. The dilation distance parameter is a vital important parameter in the banded multilevel method. If dst is too small, the algorithm may not be able to construct an optimal surface based on the surface computed from the previous level volume. If dst is too large, the computational benefits of the banded multilevel method would be reduced. In our implementation, we found that $dst = 5$ gives a good compromise between accuracy and performance for experimented volumes with different resolutions. Surface carving is applied to the banded volume and calculate the optimal surface in this level. This uncoarsening procedure is executed recursively in the successive higher resolution banded volume until the minimum cut is obtained for the original banded volume, yielding the final optimal surface, as shown the bottom right part of Fig. 5.

All volume graphs except the coarsest graph are constructed from the banded volume, and they are significantly smaller than the full volume at each level. As a result, both the memory consumption and the running time of surface carving are greatly reduced compared with a single surface carving on the original volume, and this banded multilevel method makes it possible to reduce large volumes with faster speed and less memory consumption.

3.4 Automatic data reduction

After describing surface carving for a given dimension, this section shows how to apply this method iteratively to implement automatic data reduction with the specified size or reduction rate. Given the reduced volume size, we can perform surface carving in each dimension to be reduced successively until the reduced volume size reaches the given size. For example, Fig. 3 is an volume reduction example of with the given size 242 × 236 × 56, and the original volume size is 254 × 248 × 56. In this situation, the proposed volume reduction approach is similar to volume resizing.

Besides the reduced volume size, users can also specify the reduction rate. It is a more convenient way for users and more effective for volume reduction, as the reduction process cannot only optimize the carved surface by minimizing the importance of voxels on the surface, but also choose the optimal direction with the least importance. Given the reduction rate, we automatically find which of the three directions (x, y, or z) to carve in each iteration, so that the distortion or artifact incurred is minimized. Since we aim at preserving important features while reducing the volume data, surface with the least cost $C_{\text{cut}}^{\text{least}}$ is selected and discarded, defined as follows:

$$C_{\text{cut}}^{\text{least}} = \min\{C_{\text{cut}}^{x}, C_{\text{cut}}^{y}, C_{\text{cut}}^{z}\}, \tag{5}$$

where C_{cut}^{x}, C_{cut}^{y} and C_{cut}^{z} are the minimum costs along the x, y and z directions. By iteratively carving surfaces with the least costs, we can obtain a reduced volume at the desired reduction rate. As shown in Fig. 6, under the same reduction rate, results with carving along any of the three directions only clearly show more visual artifacts (Fig. 6c–e), while the result optimized both the carved direction and surface preserves important features and detailed information (Fig. 6f–g).

It is usually difficult for users to specify the desired size or reduction rate directly, as the inappropriate parameter may result in serious visual artifacts due to over-reduction or cannot reduce the volume effectively due to under-reduction. It is, therefore, useful to indicate at what reduction rate, the reduced volume starts to include the serious distortion or clear visual artifact. Figure 7 shows the cost curve of the fuel data set, i.e., the cost in each iteration, and it also indicates the relation between the distortion incurred and the reduction rate. All important features are preserved after carving 64 surfaces, because features are completely located in the middle. As more surfaces carved, relatively less important and homogeneous regions (corresponding to the bottom part of the fuel volume) are removed. The right part of the curve shows

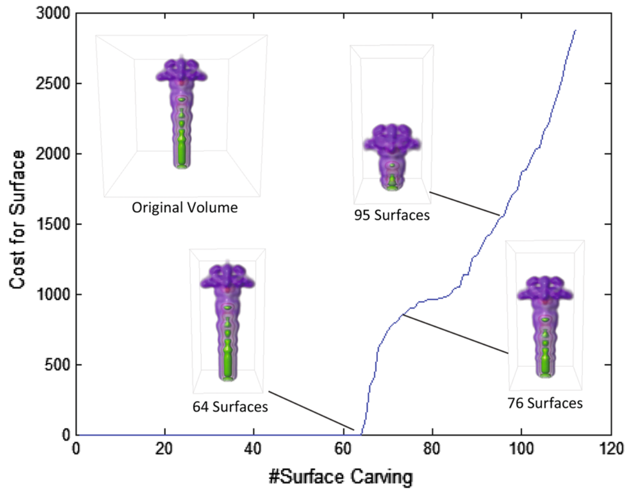


Fig. 7 Cost of each surface carving for a fuel volume

the cost for each surface carving increases rapidly, which means much more important features will be distorted, and this results in clear visual artifact. Thus, we suggest choosing the optimal reduction rates at the critical inflection of the cost curve to balance the reduction rate and reduced data quality.

4 Results and discussion

Different saliency measures can be used as the edge cost in the volume graph construction to guide surface carving. Figure 8 compares the gradient magnitude, the opacity transfer function and their combination for an atom volume. Surface carving based on the opacity transfer function carves out contextual regions on the left side shown in Fig. 8b. Figure 8c is the surface carving result based on the gradient magnitude. Since the blank regions under this transfer function have smaller gradient magnitudes, these regions are compressed during volume reduction. However, when the volume is further reduced, the relatively less important regions around the right boundary are discarded, as the gradient magnitudes in these regions are smaller than the one in the middle blank regions in the volume. The hybrid energy function based on the opacity transfer function and gradient magnitude gives the best volume reduction result shown in Fig. 8d ($\lambda = 0.1$ in Eq. 1). Contextual voxels are symmetrically removed from the left and right regions and the boundary of features are well preserved.

Figure 9 compares three reduction methods: cropping, linear downsampling, and the proposed surface carving for the foot data set, which mainly consists of tissues and bones. The specified transfer function highlights the bones by assigning large opacities to these voxels, so the saliency measure is the opacity transfer function. Compared with the original volume in Fig. 9a, the cropping result removes both tissues and bones due to its fixed geometric constraint, and the result of linear downsampling suffers from aliasing artifacts as it uniformly reduces the tissues and bones. Surface carving selectively removes voxels of tissues between bones while retaining the bones as much as possible in fitting the smaller volume size. As shown in the amplified sub-figures, the details of bones after surface carving are preserved better than the one in linear downsampling.

The volume reduction results of these three methods for a carp volume data set are shown in Fig. 10. As the user generally is interested in the bones, and the saliency value of each voxel is the opacity. Since the bones are almost filled up with the whole volume, the cropping method may remove the head or tail of the carp, and the linear downsampling method has jaggy artifacts for the thin bones in the body of the carp. Surface carving discards less important voxels on connected manifold surfaces and creates a saliency-guided reduced volume, which preserves the important features and minimizes visual artifacts.

To quantitatively measure data loss of three reduction methods, we use the following metric to compare the important information preservation between two volume data sets:

$$\frac{\sum \alpha(v')C(v')}{\sum \alpha(v)C(v)}, \tag{6}$$

where $C(v')$ and $C(v)$ are the colors of voxel v' and v obtained from the reduced and original volumes, respec-

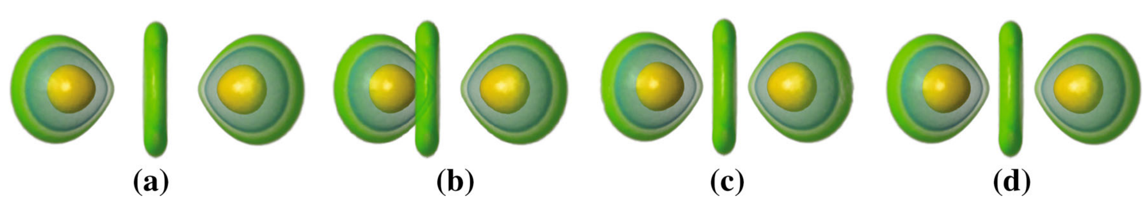


Fig. 8 This figure compares different saliency measures for a hydrogen atom volume. The volume is reduced from 128 × 128 × 128 to 92 × 128 × 128. **a** The original volume. **b** The reduced volume based on the opacity transfer function. The blank regions between two components on the left side are removed. **c** The reduced volume based on the gradient magnitude. As the blank regions under this transfer function have non-zero gradient magnitudes, the carved surfaces are symmetric and the voxels of the feature boundary on the right side are also removed due to its relatively small gradient magnitude. **d** The reduced volume based on the hybrid combination of the opacity transfer function and gradient magnitude

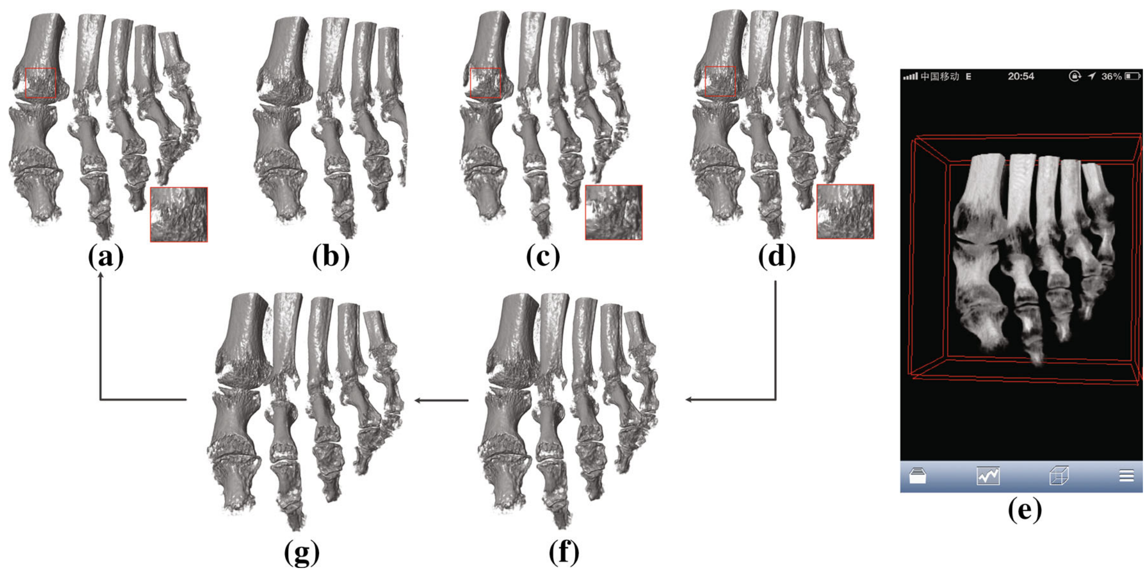


Fig. 9 The *first row* compares the cropping, linear downsampling, and surface carving for volume reduction on a foot data set. The volume is reduced from $256 \times 256 \times 256$ to $200 \times 256 \times 180$. **a** The original volume. **b** The reduced volume using the cropping method with the feature itself removed. **c** The reduced volume using the linear downsampling method with jaggy artifacts. **d** The reduced volume using the proposed surface carving. The *second row* shows the progressive volume transmission process. **e** The reduced volume (**d**) is displayed in the mobile device. **f**, **g** are two intermediate results of volume reconstruction from (**d**) to (**a**), whose size ranges from $215 \times 256 \times 200$ to $235 \times 256 \times 230$

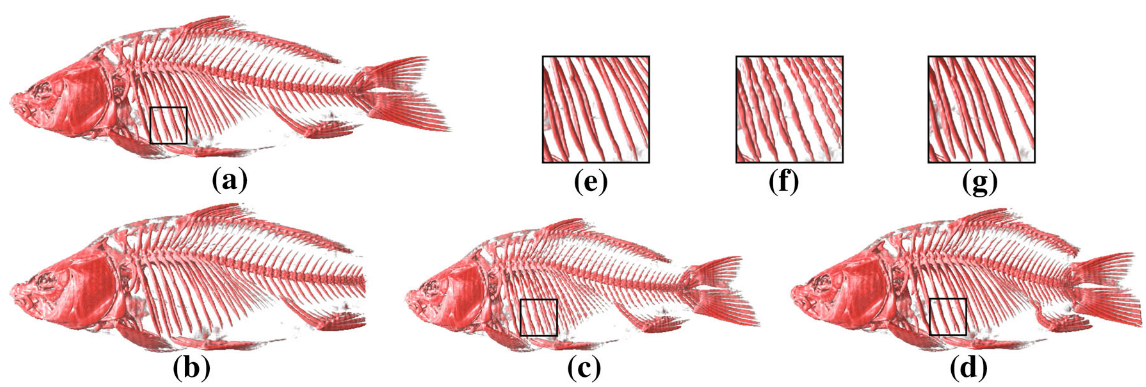


Fig. 10 The left part of this figure compares the cropping, linear downsampling, and surface carving for volume reduction on a carp data set. The volume is reduced from $512 \times 256 \times 256$ to $382 \times 256 \times 256$. **a** The original volume. **b** The reduced volume using the cropping method with the feature itself removed. **c** The reduced volume using the linear downsampling method with discontinuous artifacts. **d** The reduced volume using the proposed surface carving with important features and detail information preserved. **e**–**g** Close-ups correspond to the *black boxes* in **a**–**d**

Table 1 The visual information preservation of cropping, downsampling, and our reduced data sets

Data sets	Cropping	Downsampling	Our method
Fuel	0.1352	0.2765	0.2967
Foot	0.8360	0.4931	0.9993
Carp	0.9472	0.7478	0.9351

tively. In this metric, we multiply each voxel color by its opacity since the visual loss of a more transparent voxel is less noticeable. As shown in Table 1, our method has overall minimum data loss in terms of the visual information preservation, as surface carving explicitly considers the importance of voxels to preserve important features. The visual information preservation in our reduced carp volume at $330 \times 256 \times 256$ is less than that of the cropping method. This is because the main visual information of the carp volume under this transfer function is in its head (the tail is cropped), and the bones are uniformly distributed in the volume space (our method has to carve some bones). Figure 11 shows the relation between the visual information preservation and the reduction factor for the carp volume. Our method performs better than cropping as the reduction factor increases.

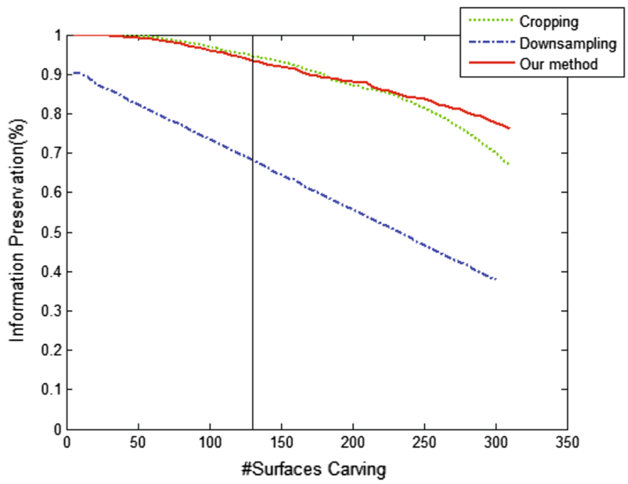


Fig. 11 The relation between the visual information preservation and the reduction factor for the carp volume. The rapid decreasing of downsampling is due to that most of the important regions are uniformly carved. When the reduction factor increases, our method performs better than the cropping method. The *black line* indicates the number of surface carving in Fig. 10

Figure 12 shows a data reduction example to obtain the optimal reduction results, when considering the visualization quality. The opacity transfer function is used for the saliency values of voxels. Similarly defined in the Fig. 7, we choose the reduction rates at the critical inflection of the cost curve, which are visualized in Fig. 12b–d. In addition, we suggest no more surface should be carved when we get the volume like Fig. 12d, for the reason that the semantic information, such as the global structure, will be destroyed.

4.1 Progressive volume transmission and reconstruction

One application of the volume reduction is progressive volume transmission and reconstruction. When a volume I is transmitted over a communication line, one would like to reduce the waiting time and display the rendered result as soon as possible. Our approach is first to reduce the original volume to produce the base volume I^n and the $n-1$ carved surfaces $\{s_i, i = 1, \dots, n-1\}$. The base volume is first transmitted, followed by the surface record s_{n-1}. Since the size of the base volume is much smaller than that of the original volume, the transmission and rendering time are largely reduced. Then, a new volume I^{n-1} is reconstructed through combining the base volume I^n with the surface s_{n-1}. This surface s_{n-1} contains all the information, such as the intensity and the 3D coordinate of each voxel, the reduction dimension. Lastly, since the surface carving is lossless, the original volume I is reconstructed exactly after all $n-1$ surface records are received. Figure 9e shows the reduced volume rendered in the mobile device and Fig. 9f–g shows two intermediate results, approximating the original volume in a progressive manner.

4.2 Performance

The performance of surface carving for first iteration with/without the banded multilevel acceleration and the size of the data sets are listed in Table 2. The data sets are illustrated in Figs. 1, 4, 6, 8, 9, 10 and 12. The performance was measured on a PC with a dual core 3.0 GHz CPU and 8GB RAM.

As the size of the coarsest volume in our banded multilevel implementation is at least $64 \times 64 \times 128$, the fuel data set has the same time for surface carving with/without acceleration. When the volume size is larger than $256 \times 128 \times 128$, it is impossible to implement surface carving due to the memory limit in a 32-bit PC. With the increasing volume size, the speedup of the banded multilevel method is more than 7.8 in our experiments. The computational time of surface carving

Fig. 12 Data reduction for a large volume data set with different reduction rates. **a** The original volume. Its size is $1024 \times 1024 \times 764$. The reduced volume with about **b** 51 % , **c** 44 % and **d** 39 % reduction rate. These reduction rates are chosen as the critical inflection of the cost curve, similarly defined in Fig. 7. No more surface should be carved for the semantic information will be destroyed, when we get the reduced volume like (**d**)

Table 2 The computation time (in seconds) of one iteration of surface carving

Data sets	Resolution	Carving time without/with acceleration	
		Without	With
Fuel	64 × 64 × 64	0.025	0.025
Balls	128 × 128 × 64	0.874	0.112
Bonsai (Fig. 6)	128 × 128 × 118	1.986	0.129
Atom	128 × 128 × 128	2.090	0.135
Vortex	128 × 128 × 128	2.714	0.312
Lobster	254 × 248 × 56	4.251	0.486
Foot	256 × 256 × 256	–	0.995
Carp	512 × 256 × 256	–	0.999
Bonsai (Fig. 12)	1024 × 1024 × 764	–	1.978

'–' Means unable to construct the volume graph due to the memory limit

depends on the voxel number of the volume, which determines the size of the volume graph, and the features in the volume, which affects the minimum energy process in the graph cut algorithm.

As can be seen from Table 2, surface carving supports interactive volume reduction for the small volume (less than 256 × 256 × 256), and it is possible to apply surface carving to the large volume data set (about 1.978 s for the 1024 × 1024 × 764 volume). When the intrinsic properties of the volume are used as the saliency measure, the original volume can be reduced in the pre-processing and the appropriate volume size can be selected for visualization based on the application.

4.3 Limitations

The proposed surface carving is designed to preserve the saliency structure of the volume during the reduction process. Maintaining the important features comes at the expense of content, and the reduction process would fail to preserve features if there is no content, i.e., all voxels are important. To solve some of those challenges, it would be helpful to specify semantics for features as constraints and guide surface carving to select less important voxels in semantic. As compared with natural images/videos, distances and relative positions between features are more important for scientific data, especially for medical data. Thus, results of surface carving may be not suitable for quantitatively visual analysis in some situations, but these results can be still used for preview.

5 Conclusion

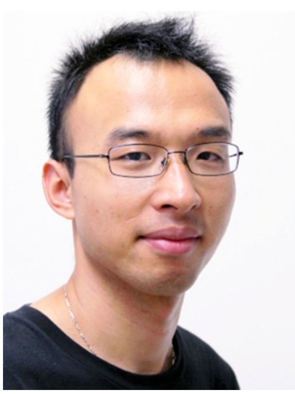
We have introduced an automatic volume data reduction method, surface carving, for interactive volume visualization. The saliency value of each voxel can be determined by the volume itself and be specified by users. Surface carving is formulated as a minimum energy problem and solved by the graph cut algorithm. It iteratively and successively removes contextual voxels while preserving important features, and generates an optimal volume at the specified size or reduction rate. We reduce large volume data sets by introducing a multilevel banded method to overcome the memory limit. Experiments demonstrate the effectiveness and efficiency of our method, and it can be applied to large volume data sets and implement data reduction for different applications. One possible future issue, we plan to investigate, is using the forward scheme [14] in the volume graph construction. Since the forward scheme has the advantage of introducing the least amount of energy into the reduced result, it will reduce distortions of well-structured features.

Acknowledgments The authors would like to thank the anonymous reviewers for their valuable comments. This work was partially supported by 863 Program Project 2012AA12A404, National Natural Science Foundation of China No. 61472354 and the Fundamental Research Funds for the Central Universities (2013QNA5010). The data sets are courtesy of Osirix DICOM, VolVis, and Siemens Medical Systems.

References

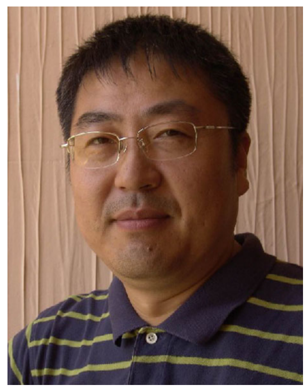
1. Moser, M., Weiskopf, D.: Interactive volume rendering on mobile devices. In: Proceedings of Vision, Modeling, and Visualization VMV '08 Conference, pp. 217–226 (2008)
2. Congote, J., Segura, A., Kabongo, L., Moreno, A., Posada, J., Ruiz, O.: Interactive visualization of volumetric data with webgl in real-time. In: Proceedings of the 16th International Conference on 3D Web Technology, Web3D '11, pp. 137–146 (2011)
3. La Mar, E.C., Hamann, B., Joy, K.I.: Multiresolution techniques for interactive texture-based volume visualization. In: Proceedings of the 10th IEEE Visualization 1999 Conference, IEEE Computer Society, (1999)
4. Weiler, M., Westermann, R., Hansen, C., Zimmermann, K., Ertl, T.: Level-of-detail volume rendering via 3d textures. In: Proceedings of the 2000 IEEE symposium on Volume visualization, VVS '00, ACM, New York, NY, USA (2000)
5. Bürger, K., Krüger, J., Westermann, R.: Direct volume editing. IEEE Trans. Vis. Comput. Graph. **14**(6), 1388–1395 (2008)
6. Yuan, X., Zhang, N., Nguyen, M.X., Chen, B.: Volume cutout. Vis. Comput. **21**(8–10), 745–754 (2005)
7. Zhao, Y., Clapworthy, G.J., Dong, F., Zhang, X., Chen, W., Viceconti, M.: Ivolbrush: an intelligent interactive tool for efficient volume selection and editing. In: Proceedings of the First International Conference on Intelligent Interactive Technologies and Multimedia, pp. 231–235 (2010)
8. Wiebel, A., Vos, F. M., Foerster, D., Hege, H.-C.: Wysiwyp: What you see is what you pick. IEEE Trans. Vis. Comput. Graph. **18**(12), 2236–2244 (2012)
9. Giachetti, A. Guitin, J.A.I., Gobbetti, E.: Edge adaptive and energy preserving volume upscaling for high quality volume rendering. In: Eurographics Italian Chapter Conference'10, pp. 17–23 (2010)
10. Wang, Q., Tao, Y., Wang, C., Dong, F., Lin, H., Clapworthy, G.: Volume upscaling using local self examples for high quality volume visualization. In: Conference on Computer-Aided Design and Computer Graphics, (2013)

11. Westermann, R.: A multiresolution framework for volume rendering. In: Proceedings of the 1994 symposium on Volume visualization, pp. 51–58 (1994)
12. Agrawal, A., Kohout, J., Clapworthy, G.J., Mcfarlane, N.J., Dong, F., Viceconti, M., Taddei, F., Testi, D.: Enabling the interactive display of large medical volume datasets by multiresolution bricking. J. Supercomput. **51**(1), 3–19 (2010)
13. Avidan, S., Shamir, A.: Seam carving for content-aware image resizing, ACM Transactions on Graphics, (Proceedings SIGGRAPH 2007) 26(3) (2007)
14. Rubinstein, M., Shamir, A., Avidan, S.: Improved seam carving for video retargeting. ACM Trans. Graph. **27**(3), 16:1–16:9 (2008)
15. Viola, I., Kanitsar, A., Groller, M.E.: Importance-driven volume rendering. In: Proceedings of the conference on Visualization '04, VIS '04, pp. 139–146 (2004)
16. Wang, Y.-S., Wang, C., Lee, T.-Y., Ma, K.-L.: Feature-preserving volume data reduction and focus+context visualization. IEEE Trans. Vis. Comput. Graph. **17**(2), 171–181 (2011)
17. Zhao, X., Zeng, W., Gu, X.D., Kaufman, A.E., Xu, W., Mueller, K.: Conformal magnifier: a focus+context technique with local shape preservation. IEEE Trans. Vis. Comput. Graph. **18**(11), 1928–1941 (2012)
18. Kindlmann, G., Durkin, J.W.: Semi-automatic generation of transfer functions for direct volume rendering. In: Proceedings of the 1998 IEEE symposium on Volume visualization, VVS '98, ACM, New York, NY, USA, (1998)
19. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. IEEE Trans. Pattern Anal. Mach. Intell. **26**(9), 1124–1137 (2004)
20. Lombaert, H., Sun, Y., Grady, L., Xu, C.: A multilevel banded graph cuts method for fast image segmentation. In: Proceedings of the Tenth IEEE International Conference on Computer Vision, IEEE Computer Society, (2005)
21. Kohli, P., Torr, P.H.S.: Dynamic graph cuts for efficient inference in markov random fields. IEEE Trans. Pattern Anal. Mach. Intell. **29**(12), 2079–2088 (2007)
22. Vineet, V., Narayanan, P.J.: Cuda cuts: Fast graph cuts on the gpu. In: Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on, pp. 1–8 (2008)



Qichao Wang received his B.Sc. and M.Sc. degrees in Computer Science and Technology from Zhejiang Gongshang University, Zhejiang, China, in 2008 and 2011, respectively. Currently he is a Ph.D. candidate in the State Key Laboratory of CAD&CG of Zhejiang University. His research interest is scientific visualization.

Yubo Tao received the B.S. and Ph.D. degrees in computer science and technology from Zhejiang University in China, in 2003 and 2009, respectively. He worked in the State Key Laboratory of CAD&CG of Zhejiang University as a Postdoctoral Researcher. Then he joined the Centre for Computer Graphics & Visualization (CCGV) at the University of Bedfordshire as a Research Fellow. He is currently an Associate Professor in the State Key Laboratory of CAD&CG of Zhejiang University. His research interests include scientific visualization and computational electromagnetics.



Hai Lin received the B.Sc., M.Sc., in electrical engineering from Xidian University, Xi'an, China, in 1987, 1990 respectively. He received the Ph.D. degree in Computer Science from Zhejiang University, Hangzhou, China in 2000. Currently, he is a Professor of Visual Computing in the State Key Lab of CAD&CG, Zhejiang University. He is also a visiting professor of the Department of Computing and Information Systems, University of Bedfordshire, UK. His research interests include computational electromagnetic, computer graphics, and scientific visualization.