Published in final edited form as:

J Math Imaging Vis. 2009 June 1; 34(2): 165–184. doi:10.1007/s10851-009-0140-7.

Digital Topology on Adaptive Octree Grids

Ying Bai^{*}, Xiao Han[†], and Jerry L. Prince^{*}

^{*} Department of Electrical and Computer Engineering Johns Hopkins University, Baltimore MD 21218

[†] CMS Software, Elekta Inc., St. Louis, MO 63043

Abstract

The theory of digital topology is used in many different image processing and computer graphics algorithms. Most of the existing theories apply to uniform cartesian grids, and they are not readily extensible to new algorithms targeting at adaptive cartesian grids. This article provides a rigorous extension of the classical digital topology framework for adaptive octree grids, including the characterization of adjacency, connected components, and simple points. Motivating examples, proofs of the major propositions, and algorithm pseudocodes are provided.

Keywords

Digital topology; adaptive octree grid; simple point characterization

1 Introduction

Digital topology [23,25,41,42] addresses the fundamental properties of binary object connectivity in two-dimensional (2D) and three-dimensional (3D) digital images. The principles and algorithms of digital topology are used in various image processing and analysis algorithms including thinning [22,31,39,43,44,52], border and surface tracing [9,27], component and tunnel counting [45], and region filling and growing [10,49]. The most widely used digital topology framework is the graph-based approach proposed by Rosenfeld [25], which divides the uniform discrete space into two graphs using a pair of adjacency¹ relations to represent the *foreground* and *background* of a binary digital image. Rules for adjacency, neighborhoods, paths, connected components, cavities, handles, and tunnels can be consistently defined on regular grids within this framework.

The concept of *simple point* was developed to describe a point (i.e., pixel or voxel) whose inclusion in either the foreground or the background does not affect the topology of the overall digital image. In other words, a simple point can be included in either the foreground or the background and neither the number of objects nor the number of handles or cavities will change. Both the theoretical characterization [6,7,12,35,44] of simple points and efficient algorithms for their detection are essential in the design of image transformations that preserve topological features. An elegant characterization of a simple point on the uniform discrete grid using only two local conditions that are easily computed was established by Bertrand et al. [6,7]. This result has found widespread application in many areas, and is particularly important in the

yingbai@jhu.edu, Xiao.Han@cmsrtp.com, prince@jhu.edu.

¹The notion of *connectivity* is sometimes used instead of *adjacency* in other literature. We use *adjacency* because it is a stronger concept, i.e. two points that are adjacent are always connected, but two points that are connected may not be adjacent.

processing of medical images, where true object topology is often known and topological correctness of the processing outcome is required [4,5,14,15,33,46].

In most computer vision and medical imaging applications, objects are represented using finite and discrete sets of pixels (in 2D) or voxels (in 3D) arranged in a rectangular coordinate system. To faithfully represent the topology of the true object, a sufficiently dense grid is required [48]. For some objects, a dense grid may be required only in some regions, while a relatively coarse grid can be used over most of the field of view. Multi-scale adaptive grids distribute the grid nodes adaptively according to the local geometry and features of the underlying object. In this way, storage requirements are lessened, efficient algorithms are possible, and detailed features of the underlying object(s), including their topology, can be preserved [11,13,34]. An illustration of this principle is provided in Fig. 1.

The definitions of adjacency and topological relations on adaptive grids are not simple extensions of those on uniform grids. Unlike a grid point on a uniform grid that is always surrounded by a fixed number of neighbor points, a grid point on an adaptive grid can have many different neighborhood configurations. This makes the study of topological properties on adaptive grids much more challenging. In this article, we present an extended digital topology framework for adaptive grids in 3D. This presentation extends and formalizes the work that we presented in recent conferences [2,3]. Our immediate target for the use of the theory presented herein is in developing multi-scale topology-preserving segmentation methods [3]. We believe that the theory and algorithms presented herein will also help promote the development of new methods for region-growing, thinning, skeletonization, and simplification (cf. [2]).

This paper is organized as follows. In Section 2, we briefly review basic concepts of digital topology for uniform grids. In Section 3, we present the extended topology framework for adaptive octree grids. In Section 4, we present and prove the extension of simple point characterization for octree grids. In Section 5, a skeleton computation algorithm on octree grids is provided as an example to demonstrate the use of our simple point characterization. Finally, in Section 6, we summarize our contributions and discuss future research directions.

2 Background: Digital Topology on Uniform Grids

In this section, we review some basic 3D digital topology concepts on standard uniform grids (see [6,7,25] for more details).

A point $x \in \mathbb{Z}^3$ is defined by (x_1, x_2, x_3) with $x_i \in \mathbb{Z}$. Three types of neighborhoods are commonly used (cf. Fig. 2(a)). These can be defined by Euclidean distance between points as follows

- 6-neighborhood: $N_6(x) = \{y \in \mathbb{Z}^3; |x_1 - y_1| + |x_2 - y_2| + |x_3 - y_3| \leq 1\}$.
- 26-neighborhood: $N_{26}(x) = \{y \in \mathbb{Z}^3; \max[|x_1 - y_1|, |x_2 - y_2|, |x_3 - y_3|] \leq 1\}$.
- 18-neighborhood: $N_{18}(x) = \{y \in \mathbb{Z}^3; |x_1 - y_1| + |x_2 - y_2| + |x_3 - y_3| \leq 2\} \cap N_{26}(x)$.

Define $N_n^*(x) = N_n(x) \setminus \{x\}$, for $n=6, 18, 26$. Two points x and y are said to be n -adjacent if $y \in N_n^*(x)$, for $n=6, 18, 26$, and denoted as $x \sim_n y$. We call the points of $N_6^*(x)$, $N_{18}^*(x) \setminus N_6^*(x)$, $N_{26}^*(x) \setminus N_{18}^*(x)$ the 6-, 18-, and 26-neighbors of x , respectively (see Fig. 2 (a)). An n -path π is a sequence of points, $x_0, \dots, x_k$, with $x_i \sim_n x_{i-1}$. π is *elementary* if all the points of the sequence are different, except for possibly $x_0 = x_k$. An elementary n -path π is called a *simple path* if each point of π is n -adjacent to only its successor (if any) and its predecessor (if any). If $x_0 = x_k$, then π is *closed*. It can be shown that any path from x_0 to x_k contains both an elementary path and a simple path from x_0 to x_k . Also, two paths

can be *concatenated* into one path when they share a common end point. For example, if $\gamma_1 = xPy$ and $\gamma_2 = yQz$ are two paths, where x, y , and z are points and P and Q are paths, then the *concatenation* of the two paths is $\gamma_1 \cdot \gamma_2 = xPyQz$.

A point set $X \subset \mathbb{Z}^3$ is *n-connected* if an *n-path* can be found between every pair of points of X . An *n-connected component* of X is a set $Y \subset X$ which is *n-connected* and no point in Y is *n-adjacent* to any other point of X . Denote $\bar{X} = \mathbb{Z}^3 \setminus X$. In order to have a correspondence between the topology of X (foreground) and that of $\bar{X}$ (background), we must use different adjacency rules for X and $\bar{X}$ [25,26,40]: if we use *n-adjacency* for X , we must use $\bar{n}$ -adjacency for $\bar{X}$, where the pair $(n, \bar{n}) \in \{(6, 26), (26, 6), (6^+, 18), (18, 6^+)\}^2$ is called a pair of consistent adjacency rules. The topology of a set $X \subset \mathbb{Z}^3$ can be characterized by its Euler number $\chi(X)$. In 3D, $\chi(X)$ equals the number of connected components in X plus the number of its cavities minus the number of its handles. A *cavity* is defined as a finite $\bar{n}$ -connected component of $\bar{X}$. A *handle* is not easy to define, but the presence of a handle in X can be detected whenever there exists a closed path in X that cannot be homotopically deformed in X to a single point without breaking the path.

The homotopy relation between closed *n-paths* [21] is a discrete analog of the homotopy defined for continuous curves in classical topology. A closed continuous curve in a set S from a base point $p \in S$ back to itself is called a *loop in S with base point p*. In other words, a loop in S with based point p is a continuous map $\gamma: [0, 1] \rightarrow S$ such that $\gamma(0) = \gamma(1) = p$. Two loops γ_1 and γ_2 in a set S that have the same base point p are said to be *homotopic (or equivalent)* in S if there exists a continuous map $h: [0, 1] \times [0, 1] \rightarrow S$ such that for all $0 \leq x \leq 1$ and $0 \leq t \leq 1$:

$$h(x, 0) = \gamma_1(x), \quad h(x, 1) = \gamma_2(x), \quad h(0, t) = h(1, t) = p.$$

In order to define the discrete analog of loop homotopy, the notion of an *elementary deformation* [7,21] in the discrete space must first be introduced. Let γ and γ' be two closed *n-paths* in X . γ is an elementary deformation of γ' in X , if they are of the form: $\gamma = pP_1u_1 \cdot P \cdot u_2P_2p$ and $\gamma' = pP_1u_1 \cdot P' \cdot u_2P_2p$, where p is the base point. u_1 and u_2 are points in X . P_1, P_2, P , and P' are paths in X , and:

- if $n = 6$, then $u_1 \cdot P \cdot u_2$ and $u_1 \cdot P' \cdot u_2$ are included in a unit square (2×2 square).
- if $n = 6^+, 18$, or 26 , then $u_1 \cdot P \cdot u_2$ and $u_1 \cdot P' \cdot u_2$ are included in a unit cube ($2 \times 2 \times 2$ cube).

Two closed *n-paths*, γ and γ' , are said to be *n-homotopic in X* if there exists a finite sequence of *n-paths*: $\gamma_0 = \gamma, \gamma_1, \dots, \gamma_m = \gamma'$, such that γ_i is an elementary deformation of γ_{i-1} in X for $i = 1, \dots, m$.

A point $x \in X$ is said to be *simple* if its deletion from X does not change the topology of the digital image, i.e., the number of connected components, the number of cavities, and the number of handles in both X and $\bar{X}$ remain constant. Whether or not a point is simple can be determined from the configuration of its immediate neighborhood ($N_8(x)$ in 2D and $N_{26}(x)$ in 3D). An efficient approach to determine whether a point is simple or not was presented in [6,7] using the concepts of *geodesic neighborhood* and *topological numbers*. For example, let $C_n(x, X)$ denote the set of all *n-connected components* of X that are *n-adjacent* to a point x , and let $\#C_n(x, X)$ denote the cardinality of $C_n(x, X)$. If we choose the adjacency pair $(n, \bar{n}) = (6, 26)$,

²Here, both 6 and 6^+ indicate conventional 6-adjacency. The use of the + superscript simply provides a convenient way to indicate that the set's complement is 18-connected. If there is no + superscript then the set's complement is 26-connected.

then the geodesic neighborhood of x relative to X is defined to be $N_{18}^*(x) \cap X$, and the corresponding topological number is $T_6(x, X) = \#C_6(x, N_{18}^*(x) \cap X)$; the geodesic neighborhood of x relative to $\bar{X}$ is defined to be $N_{26}^*(x) \cap \bar{X}$, and the corresponding topological number is $T_{26}(x, \bar{X}) = \#C_{26}(x, N_{26}^*(x) \cap \bar{X})$. These two numbers lead to a very concise characterization of a 6-simple point: $x \in X$ is 6-simple for X if and only if $T_6(x, X) = 1$ and $T_{26}(x, \bar{X}) = 1$. Some examples are shown in Figs. 2 (b)-(d).

Besides the adjacency graph approach that we have summarized above, other approaches to define topology in multidimensional regular digital space include: the oriented graph proposed by Herman [17] and Udupa [51]; and the cellular complexes approach by Kovalevsky [28], Khalimsky [20], Bertrand and Couprie [8] and Ayala et al. [1]. Extensions to higher dimensions have also been made [24,38]. Since the adjacency graph approach still enjoys the most popularity in various computer vision and image processing applications, we have currently chosen to extend this framework to adaptive grids.

3 Digital Topology on Octree Grids

3.1 Terminology and Notation

We first introduce some basic notations and concepts on octree grids.

Definition 1 An *octree grid* is a hierarchical cartesian grid. A *cell* is the basic unit of an octree grid; it is a cube with 6 faces, 12 edges, and 8 vertices. Each cell can be divided into eight child cells. The root cell represents the entire domain and is at level 0. The *resolution level* of a cell is defined by starting from the root cell and adding one for each refinement. A *leaf cell* is a cell that has no child cells.

See Fig. 3 for an illustration of an octree grid.

Definition 2 The domain of a digital image defined on an octree grid is the set Ω of all vertices of the octree leaf cells. The location of a point $P \in \Omega$ is given by an integer-valued triplet (x, y, z) , which also represents its position on the underlying finest resolution uniform grid.

Because leaf cells have different sizes, the neighborhoods on an octree grid can no longer be defined using Euclidean distance. Instead we use the following definitions.

Definition 3 Three types of *neighborhoods* are defined for each point $x \in \Omega$:

- “EDGE (E)-neighborhood” of x : $N_E(x) = \{x' \in \Omega: x \text{ and } x' \text{ are the two vertices of an edge of a leaf cell}\}$
- “SQUARE (S)-neighborhood” of x : $N_S(x) = \{x' \in \Omega: x \text{ and } x' \text{ are two of the four vertices of a square of a leaf cell}\}$
- “CUBE (C)-neighborhood” of x : $N_C(x) = \{x' \in \Omega: x \text{ and } x' \text{ are two of the eight vertices of a cube of a leaf cell}\}$

We define the E -neighbors of x to be $N_E(x)$; the S -neighbors of x to be $N_S(x) \setminus N_E(x)$, and the C -neighbors of x to be $N_C(x) \setminus N_S(x)$.

See Fig. 4 for illustrations of these neighborhoods on both regular and octree grids.

Note that the E -, S -, and C -neighborhoods on an octree grid are analogous to the 6-, 18-, and 26-neighborhoods on a uniform grid, respectively. But unlike points on a uniform grid, points at resolution transitions of an octree grid are not guaranteed to have a regular neighborhood

— they can have different numbers of neighbors in different directions. The distances between the root point and neighbor points also vary according to the sizes of the neighbor leaf cells. If all the neighbor leaf cells of the root point are of the same size, then we call it a *single-level neighborhood*; otherwise we call it a *multi-level neighborhood* (cf. Fig. 4). Now we define several terms for grid resolution transitions.

Definition 4 A *transition face* (or *face diagonal*) refers to a face (or face diagonal) of a leaf cell whose neighbor cell at the same resolution level that shares the same face (or face diagonal) is not a leaf cell. A *transition edge* refers to an edge of a leaf cell whose three neighbor cells at the same resolution level that share the same edge are not all leaf cells. Hence, a transition face (or edge, or face diagonal) consists of finer resolution faces (or edges, or face diagonals) belonging to the children of the neighbor cell(s).

See Fig. 5(a) for an example.

The notions of *adjacency* and *path* on octree grids are defined as follows.

Definition 5 Two points $x \in \Omega$ and $y \in \Omega$ are said to be α -adjacent ($\alpha \in \{E, S, C\}$), denoted as $x \sim_{\alpha} y$, on an octree grid if $y \in N_{\alpha}^*(x)$.

Definition 6 An α -path π on an octree grid is a sequence of points $x_0, x_1, \dots, x_k$, such that $x_i \sim_{\alpha} x_{i-1}$, for $1 \leq i \leq k$. If $x_0 = x_k$, then the path is called *closed*. π is *elementary* if all points of the sequence are unique, except possibly $x_0 = x_k$. An elementary α -path π is *simple* if each point of π that has a single-level neighborhood is α -adjacent to only its successor (if any) and its predecessor (if any) in π , while each point of π that has a multi-level neighborhood is α -adjacent to at most four successors (if any) and four predecessors (if any) in π ³, and the multiple successors (or predecessors) must be included in a transition edge or a transition face diagonal.

See Fig. 5(b) for an example.

Once the concepts of “adjacency” and “path” on Ω are defined, the definitions of foreground (object) X , background $\bar{X}$, connected components, and cavities are straightforward extensions of those for $\mathbb{Z}^3$ in Section 2. Defining handles on Ω requires extension of other concepts and will be discussed later. As in uniform grids, to avoid topological anomalies we must use two different kinds of adjacency, α and $\bar{\alpha}$, for X and $\bar{X}$ on octree grids, respectively, where the pair $(\alpha, \bar{\alpha}) \in \{(E, C), (C, E), (E^+, S), (S, E^+)\}$ ⁴.

3.2 Topological inconsistencies at resolution transitions

As far as topology is concerned, octree grids are fundamentally different than uniform grids because they have cells of different resolutions. Since the octree grid is used to more efficiently represent objects that would otherwise be defined on a fine resolution grid, it is a fundamental premise that object configurations on the octree grid must unambiguously specify the topology of the same object on the fine grid. On uniform grids, there are ambiguous configurations sometimes referred to as ambiguous faces and ambiguous cubes [36,37]. These types of ambiguities also exist on octree grids, and they are easily resolved by using consistent adjacency rules as mentioned earlier [19,29,30]. On an octree grid, however, there are additional configurations that can create ambiguities not resolvable by the use of consistent adjacency rules. Such configurations take place at the transitions between cells of different

³There can be four successors (or predecessors) when $\alpha = E$, and the root point has four E -neighbors in one direction, i.e., the four neighbor leaf cells that contain the four E -neighbors are at four different resolution levels.

⁴Here the + superscript is used to indicate that the set's complement is S -connected. If there is no + superscript then the set's complement is C -connected.

resolutions in an octree grid. For example, the configurations shown in Fig. 6 are problematic because the connectivity defined by the coarse resolution cell and fine resolution cells are inconsistent on the transition face/edge. Assume that the foreground (black points) and background (white points) have adjacency rule of (E, C) or (E^+, S) . In Fig. 6(a), the two white points are E -connected on the transition edge according to the coarser cell, while they are disconnected according to the two finer cells. Similarly, in Fig. 6(e), the two black points are S -connected on the transition face according to the coarser cell, while they are disconnected according to the four finer cells. Figure 6 shows the 6 basic invalid configurations at resolution transitions when the adjacency rule of (E, C) or (E^+, S) is assumed and the resolution levels of neighbor cells differ by 1. Other invalid configurations can be derived from these basic cases by reflections, symmetrical rotations, switching the role of foreground and background, or changing adjacency rules.

Such ambiguous resolution transitions are caused by the fact that the coarse grid cell at the interface is inadequate to represent the topology of the underlying object in these cases. In theory, it would be possible to resolve these ambiguities by establishing a set of new adjacency rules for each ambiguous case, which however would introduce considerable extra complexity. To resolve the ambiguities without making new rules, the coarse cell should not be used in the first place, i.e., it should be split up into finer cells to represent the additional resolution that is implied by the ambiguity. This is consistent with our premise for using adaptive grids — that is, grid nodes are adaptively distributed so that the topology of the underlying object is resolved correctly and efficiently, i.e., coarse resolution cells are used where the object geometry is simple and fine resolution cells are used where the object geometry is complex. Because of this reasoning, we disallow these types of ambiguous configurations, and refer to them as *invalid cases*. We now provide a mathematical characterization of invalid cases.

We denote the set of points included in a coarse resolution edge/face as I , and the set of points included in the corresponding fine resolution edges/faces as I' . So we must have $I \subset I'$. Now we denote the number of α -connected components in the foreground (resp. background) in a set I as $\#C_\alpha(I, X)$ (resp. $\#C_\alpha(I, \bar{X})$). Then we have the following definition of an invalid case.

Definition 7 An *invalid case* refers to a configuration on a transition edge/face, such that

$$\#C_\alpha(I, X) \neq \#C_\alpha(I', X) \text{ or } \#C_\alpha(I, \bar{X}) \neq \#C_\alpha(I', \bar{X}).$$

Fig. 6 provides some examples of invalid cases.

In order to keep a consistent definition of topology over the entire grid, invalid cases are not allowed on an octree grid. Therefore, we must not create these configurations in the generation of an octree representation of a digital object or in any further manipulation (region growing, thinning, etc.) of the object. We refer to an octree grid that has no invalid cases as a *valid octree grid* (VOG). From this point on, all theory is established and discussed for VOGs only.

3.3 Handles and deformations on VOGs

Now we will define the notion of a *handle* on a VOG. A handle is a topological structure that cannot be continuously shrunk to a point. In the discrete domain, a handle is detected whenever there exists a closed path in X that cannot be homotopically deformed to a point within the object X itself. A homotopic deformation is one in which the deforming path never breaks in order to cross over another path in $\bar{X}$. Such a deformation has been defined on uniform grids in [7,21,25]. Extension of this definition to VOGs is not straightforward due to the extra complications at resolution transitions. Now we will define two types of basic homotopic deformations of a closed path in a VOG, one for shortening or expanding the path and another for allowing the path to transit from one resolution level to another. It is easy to verify that

these deformations can never cross any path of $\bar{X}$, as will be explained later. Using a concatenation of these two basic deformations, we can then define a general homotopic deformation on VOGs.

Definition 8 Let $\gamma \in X$ and $\gamma' \in X$ be two closed paths, and choose α -adjacency for X . We say that γ' is an *elementary deformation (E-def)* of γ on VOGs, denoted as $\gamma \xrightarrow{E} \gamma'$, if γ and γ' are of the form: $\gamma = pP_1u_1 \cdot P \cdot u_2P_2p$, $\gamma' = pP_1u_1 \cdot P' \cdot u_2P_2p$, where p , u_1 , and u_2 are points, P_1 , P_2 , P , and P' are paths, and:

- if $\alpha = E$, then $u_1 \cdot P \cdot u_2$ and $u_1 \cdot P' \cdot u_2$ are included in a face of a leaf cell.
- if $\alpha = E^+$, S , or C , then $u_1 \cdot P \cdot u_2$ and $u_1 \cdot P' \cdot u_2$ are included in a cube of a leaf cell.

Let us use Fig. 7(a) to illustrate E-def's under different adjacency rules. Assume C -adjacency for the closed path $\gamma = pP_1x_1x_2x_3x_4x_5P_2p$, where p is a point and P_1 and P_2 are paths (not shown in the figure). Let $u_1 = x_1$, $u_2 = x_5$, $P = x_2x_3x_4$, and $P' = \emptyset$. Then we can deform

$\gamma \xrightarrow{E} \gamma' = pP_1x_1x_5P_2p$. Let $u_1 = x_1$, $u_2 = x_1$, $P = x_2x_3x_4x_5$, and $P' = \emptyset$. Then we can deform $\gamma \xrightarrow{E} \gamma'' = pP_1x_1P_2p$. Now assume E -adjacency for the closed path $\gamma = pP_1x_1x_6x_5x_7P_2p$. Let $u_1 = x_1$, $u_2 = x_7$, $P = x_6x_5$, and $P' = \emptyset$. Then we can deform $\gamma \xrightarrow{E} \gamma' = pP_1x_1x_7P_2p$.

The above definition of E-def is an extension of the original definition on uniform grids (cf. [7,25]). Applying an E-def on a closed path in X can either shorten or expand the path in length without crossing any path in $\bar{X}$. Since an E-def is only allowed to occur inside a leaf cell cube or a leaf cell face, there is not enough room for a path in $\bar{X}$ to be crossed. One may be concerned about an E-def that occurs in a coarse resolution cell that has transition edges/faces. Since VOGs are constrained at resolution transitions such that the coarse resolution edge/face is always topologically equivalent to the fine resolution edges/faces, there cannot exist any path in $\bar{X}$ only defined on the fine resolution edges/faces. This guarantees that an E-def in X within the coarse cell will not cross any path in $\bar{X}$. Fig. 7(b) helps to illustrate the above argument. Assume E^+ -adjacency for the black points (X), and S -adjacency for white points ($\bar{X}$). Now we want to deform the closed path $\gamma = pP_1x_1x_3x_5x_7P_2p$ inside this leaf cell with a transition face.

According to the definition of E-def, we can deform $\gamma \xrightarrow{E} \gamma' = pP_1x_1P_2p$. However, if the point x_9 is white, such an E-def would cross a path in $\bar{X}$ containing x_9 . But it turns out that such a case cannot occur because it is an invalid case forbidden on VOGs. This again demonstrates the value — indeed, necessity — of restricting our discussion to VOGs.

Now we define another type of basic deformation that allows the deformed path to transit in between different resolutions.

Definition 9 Let $\gamma \in X$ and $\gamma' \in X$ be two closed paths. We say that γ' is a *transitional deformation (T-def)* of γ on a VOG, denoted as $\gamma \xrightarrow{T} \gamma'$, if γ and γ' are of the form: $\gamma = pP_1u_1 \cdot P \cdot u_2P_2p$, $\gamma' = pP_1u_1 \cdot P' \cdot u_2P_2p$, where p , u_1 , and u_2 are points, P_1 , P_2 , P , and P' are paths, and $u_1 \cdot P \cdot u_2$ and $u_1 \cdot P' \cdot u_2$ are included inside a transition edge or a transition face diagonal.

We use Fig. 7(c) to illustrate T-def's. To let the path $\gamma = pP_1x_1x_2x_3P_2p$ transit from finer to coarser resolution, we let $u_1 = x_1$, $u_2 = x_3$, $P = x_2$, and $P' = \emptyset$, and deform

$\gamma \xrightarrow{T} \gamma' = pP_1x_1x_3P_2p$. A reverse deformation lets the path transit from coarse to fine. As another example, to let the path $\gamma = pP_1x_2x_1P_2p$ transit from finer to coarser resolution, we let $u_1 = x_2$, $u_2 = x_1$, $P = \emptyset$, and $P' = x_3$, and deform $\gamma \xrightarrow{T} \gamma' = pP_1x_2x_3x_1P_2p$. Similarly, on the transition face diagonal, we can deform the path $\gamma = pP_1x_1x_4x_5P_2p$ by $\gamma \xrightarrow{T} \gamma' = pP_1x_1x_5P_2p$; or deform

the path $\gamma \xrightarrow{T} \gamma' = pP_1x_4x_5x_1P_2p$. It should be clear that a T-def that only occurs on a transition edge/face diagonal cannot cross any path in $\bar{X}$.

Now we are ready to define homotopic deformations and handles.

Definition 10 A homotopic deformation of a closed path in a VOG is defined to be a sequence of E-def's and T-def's. A *handle* is said to exist in X when there is a closed path in X that cannot be homotopically deformed in X to a single point.

Fig. 8 shows how to detect a handle by deforming a path on VOGs. Let us assume E -adjacency for black points. Fig. 8(a) shows a closed path $\gamma = x_1x_2x_3x_4x_5x_6x_7x_1$ that does not define a handle, since we can perform the following homotopic deformations: $\gamma \xrightarrow{E} \gamma' = x_1x_4x_5x_6x_7x_1$; $\gamma' \xrightarrow{T} \gamma'' = x_1x_5x_6x_7x_1$; $\gamma'' \xrightarrow{E} \gamma''' = x_1$, to deform γ into a point. Fig. 8(b) shows a closed path $\gamma = x_1x_2x_3x_4x_5x_6x_7x_8x_9x_1$ that defines a handle, as we perform the following homotopic deformations: $\gamma \xrightarrow{E} \gamma' = x_1x_2x_3x_4x_5x_6x_7x_1$; $\gamma' \xrightarrow{T} \gamma'' = x_1x_2x_3x_4x_5x_6x_1$, and then we can no longer shorten the path γ'' to deform it to a point.

3.4 Algorithm for the generation of VOGs

In this section, we present an algorithm to generate an octree grid representation of an object from its uniform grid samples. To start, the original uniform grid is regarded as an octree grid that is at its finest resolution everywhere. The leaf cells of this octree grid are then traversed from the bottom level to the top level. At each level, the leaf cells are evaluated one-by-one to see if they can be merged without violating the VOG constraint and some other user-specified criteria (see below). If all criteria are satisfied, then the cells are coarsened, and the process is repeated until the root of the octree is reached.

The user-specified criterion we use is specifically selected to preserve the object's shape after cell merging [13]. Let the object's shape be represented by a signed distance function (cf. [47,50]). Now consider a potential coarsening in which certain grid points will be removed due to a cell merging. If the remaining nodes can linearly interpolate the signed distance function at the locations of the removed grid points to within a tolerance ε , then the merge is allowed, otherwise it is disallowed.

The algorithm to generate a VOG under this geometric criterion can be summarized as follows.

The VOG Algorithm

1. Define a status array the same size as the finest resolution data on uniform grid. Set all values in the status array to be 1, meaning that all the grid points are currently on the octree grid. (When a grid point is deleted, its value in the status array is set to be 0.) If the octree has n resolution levels, set current level to be $n - 1$.
2. Find all the parent cells at the current level whose child cells are all leaf cells, and let Q denotes the total number of such cells found.
3. For $i = 1, \dots, Q$, check the i 'th parent cell
 - a. If there is no sign change inside this parent cell, go to (e)
 - b. Check if the merging of this parent cell (i.e. delete the body point from the octree grid; delete other face/edge points if the neighbor cells sharing the face/edge are already leaf cells) leads to any invalid cases on transition faces/edges. If true, go to (e). Otherwise continue to (c).

- c. Check the geometrical criterion for this parent cell. If the error exceeds the pre-specified error bound, go to (e). Otherwise continue to (d).
- d. Allow merging inside this parent cell.
- e. Increase i , if $i > Q$, then decrease current level and go to Step 2.

Since the conditions checked inside each parent cell on the current level are not affected by any other cells of the same level on the octree grid, the output of the cell merging algorithm is unique — i.e., the order of processing cells at each level can be arbitrary. The complexity of this algorithm is $O(N^3)$, with N being the size of the data in each dimension.

There are two possible variants of the basic VOG algorithm. The first is to only consider the geometric criterion of Step 3(c) and ignore the validness criterion of Step 3(b). In this case, the output octree grid will have fewer fine resolution cells, but it will not be a VOG which means the object topology is not well defined. If we want to keep this coarser octree grid structure and still have a VOG, we would have to modify the underlying data to eliminate the features that cannot be represented by this coarser grid. Although we will be able to define the topology of the data after this modification, it may not be the same as the topology of the original data. One algorithm to fix the data without refining the octree grid works by traversing all the leaf cells, and for each leaf cell that has an invalid case, changing the sign of the data to eliminate the invalid case. The result of this algorithm may not be unique, and one may choose the order of data point visiting to ensure fewer points are modified. As the second variant, one might want to preserve the topology of the original object defined on the uniform grid during the VOG generation. Then the above algorithm can be further augmented with a topological criterion as described in [2].

4 Simple Point Characterization on VOGs

A *homotopic transformation* of a binary digital image is one that preserves the topology of all objects as well as their connectivity and nesting relationships. Since a homotopic transformation can be regarded as a sequence of deletion or addition of simple points [33], the characterization and detection of simple points is important in the design of homotopic algorithms such as thinning and region growing. In this section, we characterize simple points on VOGs and describe an efficient algorithm for their detection.

It was proven in [6,7] for uniform grids that whether a point is simple or not can be determined from two topological numbers computed from two of the point's local geodesic neighborhoods. Below, we characterize simple points on VOGs in a similar fashion. We first extend the concepts of geodesic neighborhoods and topological numbers to VOGs. We then describe an efficient algorithm for computing the topological numbers. Finally, we propose and prove a proposition that characterizes simple points on VOGs using topological numbers.

4.1 Geodesic neighborhoods and topological numbers

Topological numbers were introduced in [32] as an effective way to characterize the topological type of a given point. A topological number gives the number of connected components within a certain geodesic neighborhood around a point. Based on our new adjacency rules on VOGs, we can extend the original definitions of geodesic neighborhoods and topological numbers to VOGs.

Definition 11 Let $X \subset \Omega$ and $x \in \Omega$ where Ω is the domain of a VOG. The *geodesic neighborhood of order k* of x with respect to X is the set $N_\alpha^k(x, X)$ defined recursively by:

$$N_\alpha^1(x, X) = N_\alpha^*(x) \cap X \text{ and } N_\alpha^k(x, X) = \cup \{N_\alpha(y) \cap N_c^*(x) \cap X, y \in N_\alpha^{k-1}(x, X)\}.$$

In other words, $N_{\alpha}(x)^k(x, X)$ is the set of all the points y in $N_c^*(x) \cap X$ that there exists an α -path from x to y of length less than or equal to k . A geodesic neighborhood in a uniform grid always exists within a uniform $3 \times 3 \times 3$ neighborhood including x . In a VOG, geodesic neighborhoods can exist on multiple resolution levels, extending far beyond the capacity of a $3 \times 3 \times 3$ neighborhood around x . Taking Fig. 9(a) as an example, where we assume E^+ -adjacency for the black points(X) and S -adjacency for the white points ($\bar{X}$). The geodesic

neighborhoods $N_E^3(x, X)$ and $N_S^2(x, \bar{X})$ of the gray point x are shown in Fig. 9(b) and Fig. 9(c), respectively. Since the neighbor cells of x are at three resolution levels, there exist multiple neighbor points in the same direction that are included in a transition edge or transition face diagonal.

Definition 12 Let $X \subset \Omega$ and $x \in \Omega$ where Ω is the domain of a VOG. The *topological numbers* of the point x relative to the set X are:

$$\begin{aligned} T_E(x, X) &= \#C_E(N_E^2(x, X)) \\ T_{E^+}(x, X) &= \#C_E(N_E^3(x, X)) \\ T_S(x, X) &= \#C_S(N_S^2(x, X)) \\ T_c(x, X) &= \#C_c(N_c^1(x, X)) \end{aligned}$$

where $C_{\alpha}(Y)$ denotes the set comprising all α -connected components of Y , and $\#$ denotes the cardinality of a set. When counting connected components inside geodesic neighborhoods, we use the edges, the squares, and the cubes that belong to all the neighbor cells as well as all the transition edges/faces.

Note that the orders of different geodesic neighborhoods must be chosen properly. The guideline for choosing the order is that: when adjacency α is assumed for set X , we should choose the order k such that: (1) there cannot exist any closed α -path in $N_{\alpha}^k(x, X) \cup \{x\}$ that forms an α -handle. This means if an α -handle is removed by deletion or addition of x , the remaining of the handle cannot be one piece of connected component in $N_{\alpha}^k(x, X)$, i.e. we must have $T_{\alpha}(x, X) \geq 2$. (2) if there exists a closed α -path in $N_{\alpha}^k(x, X) \cup \{x\}$ that is not a handle, after the deletion or addition of x , the remaining of the handle should still be one connected component in $N_{\alpha}^k(x, X)$. For example, let us examine why we must choose order $k = 3$ for adjacency E^+ . If we choose order $k > 3$, then an E -handle can exist in $N_E^k(x, X) \cup \{x\}$ as shown in Figs. 10(a)–(b). On the other hand, if we choose $k < 3$, a closed path in $N_{\alpha}^k(x, X) \cup \{x\}$ that is not a handle could generate two connected components in $N_{\alpha}^k(x, X)$ when x is deleted as shown in Figs. 10(c)–(d).

Although geodesic neighborhoods on VOGs are complex when they involve points at transition edges or faces, there is a certain property that makes the computation of the topological numbers easier than it appears. Before describing this property, we first introduce the concept of a *neighbor group*.

Definition 13 If $P_1, P_2, \dots, P_n$ are multiple neighbor points of x included in a transition edge (or face diagonal) inside a geodesic neighborhood $N_{\alpha}^k(x, X)$, they form a *neighbor group* of x in $N_{\alpha}^k(x, X)$.

Since a transition edge is shared by four neighbor cells, a neighbor group on a transition edge can have at most 4 points (i.e. when all four neighbor cells are at different levels). Similarly,

since a transition face diagonal is shared by two neighbor cells, a neighbor group on a transition face diagonal can have at most 2 points (i.e. when the two neighbor cells are at different levels).

Property 1 Assume that the deletion of x from X (or the addition of x to X) does not cause any invalid cases. If $P_1, P_2, \dots, P_n$ form a neighbor group of x inside $N_\alpha^k(x, X)$, then they must be α -connected on that transition edge (or face), and there exists an α -path connecting them that does not include x . Therefore, $P_1, P_2, \dots, P_n$ belong to the same α -connected component in $N_\alpha^k(x, X)$.

See Appendix A for a proof. Note that the assumption made in this property is to ensure that the validness constraint is satisfied, since the property is only used for simple point characterization on VOGs.

Based on Property 1, points of a neighbor group are always connected inside the geodesic neighborhood, and therefore can be regarded as one point existing at multiple levels. This means that we can possibly represent geodesic neighborhoods on VOGs using only uniform arrays, on which connected components counting becomes much easier. To encode all the information of a geodesic neighborhood on VOGs, we use a uniform $3 \times 3 \times 3$ array of points or neighbor groups. This array contains 8 cubes, and each cube represents a neighbor cell of the root point. We also store the level of each neighbor cell in the center of each cube in the array.

We refer to this array as a *multi-level array* defined as follows.

Definition 14 Let n be an adjacency rule on the uniform grid that is analogous to the α -adjacency rule on the VOG (cf. Section 3). We define a *multi-level array* (MLA) form of a geodesic neighborhood $N_\alpha^k(x, X)$ on VOGs to be a $3 \times 3 \times 3$ point array (denoted as $U_n^k(x, X)$), such that: every point p in $U_n^k(x, X)$ represents either a single neighbor or a neighbor group in that particular direction in $N_\alpha^k(x, X)$. If p represents a single neighbor $q \in N_\alpha^k(x, X)$, then p is tagged with the level of the leaf neighbor cell connecting x and q . If p represents a neighbor group $Q \subset N_\alpha^k(x, X)$, then p is tagged with all the different levels of the leaf neighbor cells connecting x and points in Q . Each of the 8 cubes in the array is tagged with the level of the corresponding leaf neighbor cell.

Fig. 11 illustrates how to convert part of a geodesic neighborhood $N_s^2(x, X)$ (left figure) into its MLA form $U_{18}^2(x, X)$ (right figure). Assume the smaller cell is at resolution level 2, the larger cell is at resolution level 1. The neighbor group $\{a_1, a_2\} \subset N_s^2(x, X)$ are merged $\widehat{a} \in U_{18}^2(x, X)$. Similarly, the neighbor group $\{d_1, d_2\} \subset N_s^2(x, X)$ and $\{e_1, e_2\} \subset N_s^2(x, X)$ are merged into $\widehat{d} \in U_{18}^2(x, X)$ and $\widehat{e} \in U_{18}^2(x, X)$, respectively. Points b, c , and f in $N_s^2(x, X)$ correspond to $\widehat{b}, \widehat{c}$, and $\widehat{f}$ with tagged level information.⁵ The two cubes are also tagged with levels.

It should now be apparent that there exists a one-to-one correspondence between an α -connected component in $N_\alpha^k(x, X)$ and an n -connected component in $U_n^k(x, X)$. Therefore, we can compute $T_\alpha(x, X)$ by counting the number of n -connected components in $U_n^k(x, X)$, which is a standard computation on the uniform grid. The notions of “adjacency” and “path” in MLA

⁵In all later notations, point $\widehat{i}$ in the MLA form corresponds to point(s) i (s) in the original octree form.

forms are slightly different from those defined on uniform grids, due to the extra level information.

Definition 15 Two points p and q are n -adjacent in $U_n^k(x, X)$ (denoted as $p-U_n-q$) if they are n -adjacent in the uniform array via a cube that is tagged with a resolution level shared by both p and q .

In Fig. 11, we have $\widehat{a}-U_{18}-\widehat{b}$ through the cube tagged with level 1, as both $\widehat{a}$ and $\widehat{b}$ are tagged with level 1; whereas $\widehat{b}$ and $\widehat{f}$ are not 18-adjacent in $U_{18}^2(x, X)$ since they do not share a common level. Note that $\widehat{b}$ and $\widehat{e}$ are not 18-adjacent in $U_{18}^2(x, X)$ either, although they are both tagged with level 1, as the cube they share are tagged with level 2. Paths in $U_n^k(x, X)$ are defined accordingly using this modified adjacency concept. Topological numbers on a VOG can now be defined using the MLAs, as follows:

$$\begin{aligned} T_e(x, X) &= \#C_6(U_6^2(x, X)) \\ T_{e^+}(x, X) &= \#C_6(U_6^3(x, X)) \\ T_s(x, X) &= \#C_{18}(U_{18}^2(x, X)) \\ T_c(x, X) &= \#C_{26}(U_{26}^1(x, X)) \end{aligned}$$

Figs. 9(d)–(e) demonstrate how to convert the two geodesic neighborhoods in Figs. 9(b)–(c) into MLA forms for computing the topological numbers.

The advantage of using $U_n^k(x, X)$ for computing topological numbers is that resolution transition (i.e., counting along a transition edge or face) is implicitly handled by counting through a multi-level point in $U_n^k(x, X)$. Furthermore, it becomes more convenient to describe homotopic deformations of a closed path inside a local neighborhood of x . Only one type of basic deformation — the elementary deformation (E-def) — is needed to describe a homotopic deformation inside an MLA. T-def's are no longer necessary since the resolution transition of a path is also implicitly handled by E-def's through multi-level points. Denote

$\tilde{N}_\alpha^k(x, X) = N_\alpha^k(x, X) \cup \{x\}$ and $\tilde{U}_n^k(x, X) = U_n^k(x, X) \cup \{x\}$. The definition of an E-def inside $\tilde{U}_n^k(x, X)$ is similar to that on uniform grids, except the underlying concepts of “adjacency” and “path” are modified.

Definition 16 Let $\gamma \in X$ and $\gamma' \in X$ be two closed paths in $\tilde{U}_n^k(x, X)$. We say that γ' is an

elementary deformation (E-def) of γ in $\tilde{U}_n^k(x, X)$ (denoted as $\gamma \xrightarrow{\widehat{E}} \gamma'$) if γ and γ' are of the form: $\gamma = pP_1u_1 \cdot P \cdot u_2P_2p$, $\gamma' = pP_1u_1 \cdot P' \cdot u_2P_2p$, where p , u_1 , and u_2 are points, P_1 , P_2 , P , and P' are paths in $\tilde{U}_n^k(x, X)$, and:

- if $n = 6$, then $u_1 \cdot P \cdot u_2$ and $u_1 \cdot P' \cdot u_2$ are included in a unit square (2×2 square).
- if $n = 6^+$, 18, or 26, then $u_1 \cdot P \cdot u_2$ and $u_1 \cdot P' \cdot u_2$ are included in a unit cube ($2 \times 2 \times 2$ cube).

The relationship between a deformation in $\tilde{N}_\alpha^k(x, X)$ and a deformation in $\tilde{U}_n^k(x, X)$ is clear: applying to a closed path γ a sequence of E-def's and T-def's within $\tilde{N}_\alpha^k(x, X)$ is equivalent to applying to γ a sequence of E-def's within $\tilde{U}_n^k(x, X)$. Fig. 12 illustrates this equivalence. Let us

use E -adjacency for a closed path γ . Fig. 12(a) shows $\tilde{N}_\alpha^k(x, X)$, and Fig. 12(b) shows the MLA form of $\gamma = \pi_1 \widehat{abxcde} \pi_2$ in $\tilde{U}_n^k(x, X)$. If we apply the following deformation in (a): $\gamma \xrightarrow{E} \gamma_1 = \pi_1 axc_1 c_2 de \pi_2$; $\gamma_1 \xrightarrow{T} \gamma_2 = \pi_1 axc_2 de \pi_2$; $\gamma_2 \xrightarrow{E} \gamma' = \pi_1 axe \pi_2$, it is equivalent to apply the following deformation in (b): $\gamma \xrightarrow{\bar{E}} \gamma_1 = \pi_1 \widehat{axcde} \pi_2$; $\gamma_1 \xrightarrow{\bar{E}} \gamma' = \pi_1 \widehat{axe} \pi_2$. As we shall see, this equivalence is very useful in proofs related to simple point characterization.

4.2 Characterization of simple points on VOGs

Now we will state and prove a proposition that gives a characterization of simple points on a VOG.

Proposition 1 A point $x \in X$ on a VOG is simple if and only if (1) after its removal the grid is still a VOG and (2) $T_\alpha(x, X) = 1$, $T_{\bar{\alpha}}(x, \bar{X}) = 1$, where $(\alpha, \bar{\alpha})$ is (E, C) or (C, E) or (E^+, S) or (S, E^+) .

Before proceeding to the proof, it is necessary to establish the following lemmas.

Lemma 1 Let $x \in X$ and suppose we delete x from X . Then $T_\alpha(x, X) = 0 \Leftrightarrow$ an α -component of X is removed. (See Appendix B for a proof.)

Lemma 2 Let $x \in X$ and suppose we delete x from X . Then $T_\alpha(x, X) \geq 2 \Leftrightarrow$ an α -component of X is created or an α -handle of X is removed. (See Appendix C for a proof.)

Since the handles of X correspond to the handles of $\bar{X}$, Lemma 1 and Lemma 2 lead to:

Lemma 3 Let $x \in X$ and suppose we delete x from X . Then $T_\alpha(x, X) = 1 \Leftrightarrow$ the α -components of X are preserved, no α -handles of X are removed, and no $\bar{\alpha}$ -handles of $\bar{X}$ are removed.

The reverse operation of the deletion of $x \in X$ from X is the addition of $x \in \bar{X}$ to X . The removal of handles corresponds to the creation of handles in this reverse operation. Therefore we have the following lemma:

Lemma 4 Let $x \in \bar{X}$ and suppose we add x to X . Then $T_\alpha(x, X) = 1 \Leftrightarrow$ the α -components of X are preserved, no α -handles of X are created, and no $\bar{\alpha}$ -handles of $\bar{X}$ are created.

Since that the addition of x to X is the same as the deletion of x from $\bar{X}$, and by replacing X by $\bar{X}$, by $\bar{\alpha}$, we have:

Lemma 5 Let $x \in X$ and suppose we delete x from X . Then $\bar{\alpha}$ -components of $\bar{X}$ are preserved, no $\bar{\alpha}$ -handles of $\bar{X}$ are created, and no α -handles of X are created.

Now Proposition 1 can be readily proved as follows:

Proof: If x is removed from X , then only the following three types of topological changes can happen:

1. an α -component of X is created or removed.
2. an $\bar{\alpha}$ -component of $\bar{X}$ (i.e., $\bar{\alpha}$ -cavity of X) is created or removed.
3. an α -handle of X (or an $\bar{\alpha}$ -handle of $\bar{X}$) is created or removed. (Note: an $\bar{\alpha}$ -handle of $\bar{X}$ is dual to an α -handle of X .)

From Lemma 3, we know that if $T_\alpha(x, X) = 1$, then the α -components of X are preserved, no α -handles of X are removed, and no $\bar{\alpha}$ -handles of $\bar{X}$ are removed. From Lemma 5, we know that if $T_{\bar{\alpha}}(x, \bar{X}) = 1$, then the $\bar{\alpha}$ -components of $\bar{X}$ are preserved, no $\bar{\alpha}$ -handles of $\bar{X}$ are created, and no α -handles of X are created. Therefore, if $T_\alpha(x, X) = 1$ and $T_{\bar{\alpha}}(x, \bar{X}) = 1$, no topological change will occur if x is removed from X . We can conclude that Proposition 1 is a characterization of simple points on VOGs.

4.3 Pseudocode for simple point detection on VOGs

Based on the above proposition, a simple point detection algorithm on a VOG consists of the following steps:

1. Ensure that the removal of a point causes no invalid cases.
2. Construct the two geodesic neighborhoods in MLA form.
3. Compute the two topological numbers and check if they are both 1.

Algorithm 1 provides a pseudocode that returns TRUE if the removal of x from X yields a valid new configuration. This algorithm works no matter X is foreground or background and also works for arbitrary adjacency rules. Algorithm 2 provides a pseudocode for the construction of the geodesic neighborhood $U_6^3(x, X)$ within a VOG, where it is assumed that the adjacency rule $\alpha = E^+$; geodesic neighborhoods for other adjacency rules can be constructed similarly.

Algorithm 3 provides a pseudocode to count the number of n -components in $U_n^k(x, X)$. The topological numbers can be computed using this algorithm, from which it is straightforward to determine whether a point in a VOG is a simple point or not.

5 Example: Computing Skeletons on VOGs

In this section, we present an example, skeleton computation, which is a potentially useful application of the digital topology theory on VOGs. Skeletons are widely used as efficient shape descriptors in object representation. They capture the topology and shape information of an object in a simple form and can be very useful in shape modeling, shape analysis, and etc. As volumetric data increases in size with modern technology, more efficient schemes such as adaptive grid techniques are adopted to keep the computation cost manageable. In the following, we present a method to compute skeletons on VOGs.

The first step is to generate a VOG representation of a volumetric model defined on uniform grid. The grid refinement criterion should be designed according to the purpose of the skeletonization algorithm. The design of an optimal grid adaptation criterion for skeleton computation is beyond the scope of this paper. In general, however, since the goal is to compute the skeleton of an object, the VOG should have fine resolution cells in the skeleton regions (i.e. the center of the object) and coarse cells elsewhere. In this example, we estimate the skeleton location of a binary object by first computing a distance map D to the object boundary, and then detect the regions where $|D| \ll 1$. Points in these regions are roughly at the center of the object. The VOG generation is then guided by this skeleton measure, i.e., the VOG should have higher resolution where $|D|$ is small and lower resolution where $|D|$ is close to 1.

We use a toy phantom to demonstrate how VOGs are generated for skeleton computation. Fig. 13(a) shows the triangulated boundary surface of a volumetric chair phantom defined on a uniform grid. Fig. 13(b) shows a cross section view of the computed skeleton measure. Fig. 13(c) shows a cross section view of the generated VOG overlaid on the skeleton measure. Dark regions denote the skeleton regions where $|D|$ is small, and therefore has finer resolution cells.

Fig. 13(d) shows the triangulated boundary surface of the phantom defined on the generated VOG. The VOG has only 13520 grid points (with 9284 points inside the object), whereas the original uniform grid has $128 \times 128 \times 128$ grid points (with 64116 points inside the object). Clearly, skeleton computation that involves a thinning process can be more efficiently carried out on VOGs thanks to the simplified object representation.

To compute skeleton on a VOG, we start from sorting all the object points based on two criteria: (1) points are first sorted by the distance to the object boundary; (2) points that are equidistant to the boundary are further sorted by the number of E -connected neighbors (neighbor groups) of them. The second criterion is adapted from the method proposed by Ju et al. [18] on uniform grids. It helps to generate more regular-shaped skeleton. After sorting, we apply a topology-preserving thinning to the object on the VOG. The simple point characterization that we have derived in Section 4 must be integrated into the thinning process to guarantee that only simple points are removed so that the skeleton has the correct topology. During thinning, we also preserve surface-end and curve-end points using criteria adapted from [18]. Figs. 13(e) and (f) show the triangulated boundary surfaces of the computed skeletons on the uniform grid and the VOG, respectively. Note that in Fig. 13(f), there are larger triangles at both ends of the “neck” connecting the bottom cushion and the back cushion. This is because the skeleton measure is less prominent in those regions (see Fig. 13(b)) and therefore the grid resolution there is relatively lower. The Euler number is verified to be equal to 2 for each of the boundary surface models in Figs. 13(a), and (d)–(f). We conclude that for both skeleton results on the uniform grid and the VOG, topology has been correctly preserved during thinning.

6 Conclusion

A new digital topology framework for adaptive octree grids was presented. Due to potential topological ambiguities between an object defined on an octree grid and the same object defined on the finest grain grid, we defined the concept of a valid octree grid (VOG), which contains no scale-related topological ambiguities. Fundamental concepts such as neighborhoods, adjacency, connected components, and handles are all extended to VOGs. We also developed a characterization of simple points on VOGs and provided pseudocodes for the basic steps required to detect simple points on VOGs. This new framework should be useful in the design of topology preserving algorithms on adaptive octree grids that are relevant in applications such as computer vision, computer graphics, and image processing and analysis. Future work include formulating an extension to general octree grids that do not require a validity constraint and exploring connections to algebraic topology [16].

Acknowledgments

This research was supported in part by NIH/NINDS grant R01NS037747.

This work was supported in part by NIH/NINDS Grant R01NS37747.

APPENDICES

A. Proof of Property 1

Proof:

Fig. 14, which shows the upper four neighbor cells of x , helps to illustrate the proof. There are two cases, (a) and (b), to consider.

- a. We first prove the case when the neighbor points of x in $N_\alpha^k(x, X)$ are included in a transition edge — e.g., points a , b , and d in Fig. 14. Consider a coarse leaf cell edge

(e.g., the edge xd in Fig.14). It is obvious that at most one α -component of X can exist on a coarse edge. In addition, since the topology of X defined by the finer edges must be equivalent to that by the coarser edge on any transition edge of a VOG, there can exist at most one α -component on the finer edges (e.g. the edges xa , ab and bd in Fig. 14). Therefore these multiple neighbor points must all belong to the same α -component (denoted as I). This must be true before and after the deletion of x from X (or the addition of x to X), since it is assumed that the octree grid remains a VOG before and after the change at x . Hence there exists an α -path not including x connecting these neighbor points on that transition edge. Note that if one of the fine edges (e.g. the edge bd in Fig.14) is also a transition edge containing still finer edges (e.g. the edges bc and cd in Fig.14), then we must have all the points in that transition edge belong to I (i.e. we must have $c \in I$ in Fig.14) by the validness constraint of VOGs, and the above argument is still true.

- b. Now we prove the case when the neighbor points of x in $N_\alpha^k(x, X)$ are included in a transition face diagonal — e.g., points g and i in Fig. 14. We note that there can be only two neighbor points in such a neighbor group, since x can have at most two neighbors in a diagonal direction. Consider a coarse leaf cell face (e.g. the face $xdgh$ in Fig. 14). If $\alpha = C$ or $\alpha = S$, then it is obvious that there can exist at most one α -component on that face; in this case, the proof is analogous to that of case (a). If $\alpha = E$ or $\alpha = E^+$ for X , then it is obvious that there can exist at most two α -components on the coarser face. If there is only one α -component, then the argument of case (a) holds. If there are two α -components (I_1 and I_2) in X , then we must have two points on the coarse face diagonal with each belonging to one of the two components (e.g. in Fig. 14, if g belongs to I_1 , then x must belong to I_2 , and h and d must belong to $\bar{X}$). By the validness constraint, I_1 and I_2 must be the only two α -components on the finer faces on the transition face. It follows that the neighbor of x on the finer face diagonal (e.g. the point i in Fig. 14) must be α -connected to either x or the neighbor of x on the coarser face diagonal (e.g. the point g in Fig.14) on the transition face. If it is the first case, then the deletion of x will cause an invalid case, which is not allowed on a VOG. Therefore, the neighbor of x on the finer face diagonal must be connected to the neighbor of x on the coarser face diagonal — i.e., there exists an α -path in the transition face connecting the neighbor group on the transition face diagonal.

From cases (a) and (b) we conclude that the multiple neighbor points included in a transition edge/face diagonal inside a geodesic neighborhood are always α -connected on that transition edge/face, which proves the property.

B. Proof of Lemma 1

Suppose x is deleted from X . If $T_\alpha(x, X) = 0$, then x has no α -neighbor in X , which means x is an isolated point that forms an α -component by itself. Therefore deleting x from X removes an α -component of X . On the other hand, if the deletion of x removes an α -component of X , then x must have no α -neighbors in X , which means $T_\alpha(x, X) = 0$. Fig. 15 shows some examples of an α -isolated point. In these examples, $T_\alpha(x, X) = 0$, and the point x forms a connected component by itself. Deleting x from X removes an α -component from X .

C. Proof of Lemma 2

C1. Necessary condition: Suppose x is deleted from X . If an α -component of X is created or an α -handle of X is removed, then $T_\alpha(x, X) \geq 2$.

Proof:

1. If a new α -component of X is created when x is deleted from X , it must be the case that an original α -component is separated into two components (I_1 and I_2) by the deletion of x . It follows that x is locally α -adjacent to I_1 and I_2 , and I_1 and I_2 are not connected in $X \setminus x$. Thus we must have $T_\alpha(x, X) \geq 2$.
2. If a handle is removed, then before the deletion of x , there must exist a closed simple α -path γ that contains x and has the form of $\pi_1 x_1 x x_2 \pi_2$ where π_1 and π_2 are paths and x_1 and x_2 are distinct points in $N_C(x, X)$. We denote the corresponding points of x_1 and x_2 in $U_n^k(x, X)$ by $\widehat{x}_1$ and $\widehat{x}_2$, respectively.

We now proceed to prove C1 by contradiction. Assume that $T_\alpha(x, X) \leq 1$. If $T_\alpha(x, X) = 0$, then it reduces to the case in Lemma 1. Therefore, x must have at least one α -adjacent neighbor in X and we must have $T_\alpha(x, X) = 1$ — i.e., there exists an n -path $\widehat{\pi} = \widehat{x}_1 \widehat{y}_1 \cdots \widehat{y}_l \widehat{x}_2$ in $U_n^k(x, X)$ connecting $\widehat{x}_1$ and $\widehat{x}_2$. Here $\widehat{y}_1$ is an n -neighbor of $\widehat{x}_1$ in $U_n^k(x, X)$, $\widehat{y}_l$ is an n -neighbor of $\widehat{x}_2$ in $U_n^k(x, X)$, and $\widehat{y}_i$ is an n -neighbor of $\widehat{y}_{i-1}$ in $U_n^k(x, X)$, for $i = 2 \cdots l$. We will show later that we can perform the following sequence of E-def's to γ in

$U_n^k(x, X) : \gamma \xrightarrow{\widehat{E}} \cdots \xrightarrow{\widehat{E}} \gamma' = \pi_1 \widehat{x}_1 \widehat{y}_1 \widehat{y}_2 \cdots \widehat{y}_l \widehat{x}_2 \pi_2 \xrightarrow{\widehat{E}} \gamma'' = \pi_1 \widehat{x}_1 \widehat{y}_1 \widehat{y}_2 \cdots \widehat{y}_l \widehat{x}_2 \pi_2 = \pi_1 \widehat{\pi} \pi_2$. This means that we can homotopically insert the path $\widehat{\pi}$ into γ , and can conclude that the deletion of x from X cannot remove the handle γ (since $\widehat{x}_1$ and $\widehat{x}_2$ are still connected inside $U_n^k(x, X)$ through $\widehat{\pi}$ after the deletion of x). This contradicts our assumption, and therefore we must have $T_\alpha(x, X) \geq 2$.

Now what remains to be proven is that we can deform γ to γ' using a sequence of E-def's in $U_n^k(x, X)$ under different choices of adjacency relations. We consider each case separately in the following:

- a. If $\alpha = \mathbf{C}$, then $\widehat{\pi} = \widehat{x}_1 \widehat{y}_1 \cdots \widehat{y}_l \widehat{x}_2$ is a 26-path in $U_{26}^1(x, X)$. Since $\widehat{y}_1 - U_{26} - \widehat{x}_1$ and $\widehat{x}_1 - U_{26} - x$, we must have $x, \widehat{x}_1$, and $\widehat{y}_1$ belong to a unit cube in $U_{26}^1(x, X)$. Since x always shares a common level with any point in $U_{26}^1(x, X)$, we also have $\widehat{y}_1 - U_{26} - x$. Therefore we can deform $\gamma \xrightarrow{\widehat{E}} \gamma_1 = \pi_1 \widehat{x}_1 \widehat{y}_1 x \widehat{x}_2 \pi_2$. This operation can be iterated until $\gamma_{l-1} \xrightarrow{\widehat{E}} \gamma_l = \pi_1 \widehat{x}_1 \widehat{y}_1 \widehat{y}_2 \cdots \widehat{y}_l \widehat{x}_2 \pi_2$.
- b. If $\alpha = \mathbf{S}$, then $\widehat{\pi} = \widehat{x}_1 \widehat{y}_1 \cdots \widehat{y}_l \widehat{x}_2$ is an 18-path in $U_{18}^2(x, X)$. Since $\widehat{y}_1 - U_{18} - \widehat{x}_1$ and $\widehat{x}_1 - U_{18} - x$, we must have $x, \widehat{x}_1, \widehat{y}_1$ belonging to a unit cube in $U_{18}^2(x, X)$. Now we iterate the following operation relative to x in $U_{18}^2(x, X)$:
 - if $\widehat{y}_1 - U_{18} - x$ (see Fig. 16(a) for an example), we can deform $\gamma \xrightarrow{\widehat{E}} \gamma_1 = \pi_1 \widehat{x}_1 \widehat{y}_1 x \widehat{x}_2 \pi_2$.
 - otherwise $\widehat{y}_1$ is a 26-neighbor of x , then it must be true that $\widehat{y}_2 - U_{18} - x$ (see Fig. 16(b) for an example) and $x, \widehat{x}_1, \widehat{y}_1, \widehat{y}_2$ must belong to a unit cube in $U_{18}^2(x, X)$. Hence, we can deform $\gamma \xrightarrow{\widehat{E}} \gamma_1 = \pi_1 \widehat{x}_1 \widehat{y}_1 \widehat{y}_2 x \widehat{x}_2 \pi_2$.

By iterating the above operation, we can deform γ to γ' through a sequence of E-def's.

- c. If $\alpha = \mathbf{E}^+$, then $\widehat{\pi} = \widehat{x}_1 \widehat{y}_1 \cdots \widehat{y}_l \widehat{x}_2$ is a 6-path in $U_6^3(x, X)$. It is necessary that $\widehat{x}_1$ is a 6-neighbor of x , the point $\widehat{y}_1$ is necessarily an 18-neighbor, and $\widehat{y}_2$ is either a 6-neighbor (see Fig. 16(c) for an example) or a 26-neighbor (see Fig. 16(d) for an example). We can then iterate the following operation relative to x in $U_6^3(x, X)$:

- if $\widehat{y}_2$ is a 6-neighbor of x , we can deform $\gamma \xrightarrow{\widehat{E}} \gamma_1 = \pi_1 \widehat{x}_1 \widehat{y}_1 \widehat{y}_2 x \widehat{x}_2 \pi_2$.
- otherwise, $\widehat{y}_2$ is a 26-neighbor of x . Since $\widehat{\pi}$ is simple, $\widehat{y}_3$ is necessarily a 6-neighbor of $\widehat{y}_2$ which is a 26-neighbor of $\widehat{x}_1$ (see Fig. 16(d) for an example). It is necessary that we have one of the two common 6-neighbors of x and $\widehat{y}_3$ (e.g., a and b in Fig. 16(d)) belonging to X , and it must also share a common level with $\widehat{y}_3$. Otherwise the shortest path included in X from $\widehat{y}_3$ to x would be of length 4 and $\widehat{y}_3$ could not belong to $U_6^3(x, X)$. Let us denote $\widehat{u}$ to be such a common neighbor belonging to X . Thus we can deform $\gamma \xrightarrow{\widehat{E}} \gamma_1 = \pi_1 \widehat{x}_1 \widehat{y}_1 \widehat{y}_2 \widehat{y}_3 \widehat{u} x \widehat{x}_2 \pi_2$.

By iterating the above operation, we can deform γ to γ' through a sequence of E-def's.

- d. If $\alpha = \mathbf{E}$, then $\widehat{\pi} = \widehat{x}_1 \widehat{y}_1 \cdots \widehat{y}_l \widehat{x}_2$ is a 6-path in $U_6^2(x, X)$. Since $\widehat{x}_1$ is a 6-neighbor of x , it is necessary that $x, \widehat{x}_1, \widehat{y}_1$, and $\widehat{y}_2$ are included in a unit square in $U_6^2(x, X)$. Thus we can deform $\gamma \xrightarrow{\widehat{E}} \gamma_1 = \pi_1 \widehat{x}_1 \widehat{y}_1 \widehat{y}_2 x \widehat{x}_2 \pi_2$. By iterating the above operation, we can deform γ to γ' through a sequence of E-def's.

C2. Sufficient condition: suppose x is deleted from X . If $T_\alpha(x, X) \geq 2$, then an α -component of X is created or an α -handle of X is removed.

Proof:

If $T_\alpha(x, X) \geq 2$, then there exist at least two disconnected α -components I and I_2 in $N_\alpha^k(x, X)$.

1. If I_1 and I_2 are not connected in $X \setminus \{x\}$, then the deletion of x will separate I_1 and I_2 and a new α -component will be created.
2. If I_1 and I_2 are connected by an α -path in $X \setminus \{x\}$, then there exists in X a closed simple α -path $\gamma_0 = \pi_1 c_1 x c_2 \pi_2$ connecting I_1 and I_2 , where c_1 is a point in I_1 and c_2 is a point in I_2 . Using the MLA form of the geodesic neighborhood, we have $\gamma_0 = \pi_1 \widehat{c}_1 x \widehat{c}_2 \pi_2$ connecting two n -components $\widehat{I}_1$ and $\widehat{I}_2$ in $U_n^k(x, X)$. Let γ be any closed path that is a homotopic deformation of γ_0 . We denote $\#(\gamma, \widehat{I}_1)$ as the number of occurrences of the form $x \widehat{c}_1$ or $\widehat{c}_1 x$ appearing in γ . If it can be proven that $\#(\gamma, \widehat{I}_1)$ is necessarily an odd number, then it follows that the closed path γ_0 must define a handle, since otherwise it could be deformed to a closed path γ which is a single point and we would have $\#(\gamma, \widehat{I}_1) = 0$. Furthermore, the handle defined by γ_0 is removed by the deletion of x , since otherwise γ_0 could be deformed to a closed path γ that does not contain x and again we would have $\#(\gamma, \widehat{I}_1) = 0$. We see that the above two conclusions are exactly what we intend to prove (i.e., the deletion of x removes an α -handle in X).

We prove the result that $\#(\gamma, \widehat{I}_1)$ is always an odd number by induction. Suppose γ_l is obtained from γ_0 by L consecutive E-def's or T-def's, where L is any integer. We prove that $\#(\gamma_l, \widehat{I}_1)$ is an odd number for $0 \leq l \leq L$.

When $l = 0$, we have $\#(\gamma_0, \widehat{I}_1) = 1$ since γ_0 is a simple path. Assume the property is true until $l - 1$, i.e., $\#(\gamma_{l-1}, \widehat{I}_1)$ is an odd number. Now consider $\#(\gamma_l, \widehat{I}_1)$. If the basic deformation from γ_{l-1} to γ_l does not involve x , then $\#(\gamma_l, \widehat{I}_1) = \#(\gamma_{l-1}, \widehat{I}_1)$ remains odd. Let us consider a basic

deformation (an E -def or a T -def) that involves x . It is obvious that such a deformation can only occur inside $N_c(x) \cap X = N_c^1(x, X)$, or in the MLA form $U_{26}^1(x, X)$. Consider the E -def $\gamma_{l-1} \xrightarrow{\widehat{E}} \gamma_l$ in $U_{26}^1(x, X)$ involving x :

$$\gamma_l = pP_1\widehat{u}_1 \cdot \widehat{P} \cdot \widehat{u}_2 P_2 p, \quad \gamma_{l-1} = pP_1\widehat{u}_1 \cdot \widehat{P}' \cdot \widehat{u}_2 P_2 p$$

when $n=6^+, 18, 26$, $\widehat{u}_1 \cdot \widehat{P} \cdot \widehat{u}_2$ and $\widehat{u}_1 \cdot \widehat{P}' \cdot \widehat{u}_2$ are included in a unit cube in $U_{26}^1(x, X)$; when $n=6$, $\widehat{u}_1 \cdot \widehat{P} \cdot \widehat{u}_2$ and $\widehat{u}_1 \cdot \widehat{P}' \cdot \widehat{u}_2$ are included in a unit square in $U_{26}^1(x, X)$. Since x is involved in this E -def, we must have $x \in \widehat{u}_1 \cdot \widehat{P} \cdot \widehat{u}_2$ or $x \in \widehat{u}_1 \cdot \widehat{P}' \cdot \widehat{u}_2$, thus we can decompose $\widehat{u}_1 \cdot \widehat{P} \cdot \widehat{u}_2$ and $\widehat{u}_1 \cdot \widehat{P}' \cdot \widehat{u}_2$ into a sequence of loops containing x .

$$\widehat{u}_1 \cdot \widehat{P} \cdot \widehat{u}_2 = \widehat{u}_1 \cdot \widehat{x}_1 \cdot \widehat{Q}_1 \cdot \widehat{x}_1 x \widehat{x}_2 \cdot \widehat{Q}_2 \cdot \widehat{x}_2 x \cdots x \widehat{x}_i \cdot \widehat{Q}_i \cdot \widehat{x}_i x \widehat{x}_{i+1} \cdot \widehat{Q}_{i+1} \cdot \widehat{x}_{i+1} \cdot \widehat{u}_2$$

with $x \notin \widehat{x}_1 \cdot \widehat{Q}_1 \cdot \widehat{x}_1, \dots, \widehat{x}_{i+1} \cdot \widehat{Q}_{i+1} \cdot \widehat{x}_{i+1}$, and

$$\widehat{u}_1 \cdot \widehat{P}' \cdot \widehat{u}_2 = \widehat{u}_1 \cdot \widehat{y}_1 \cdot \widehat{R}_1 \cdot \widehat{y}_1 x \widehat{y}_2 \cdot \widehat{R}_2 \cdot \widehat{y}_2 x \cdots x \widehat{y}_j \cdot \widehat{R}_j \cdot \widehat{y}_j x \widehat{y}_{j+1} \cdots \widehat{R}_{j+1} \cdot \widehat{y}_{j+1} \cdot \widehat{u}_2$$

with $x \notin \widehat{y}_1 \cdot \widehat{R}_1 \cdot \widehat{y}_1, \dots, \widehat{y}_{j+1} \cdot \widehat{R}_{j+1} \cdot \widehat{y}_{j+1}$. If $i=0$ (or $j=0$), it means that $\widehat{P}$ (or $\widehat{Q}$) does not contain x .

In the following, we will show that $\#(\gamma_l, \widehat{C}_1)$ can be obtained from $\#(\gamma_{l-1}, \widehat{C}_1)$ by adding and subtracting even numbers.

a. If $i > 0$ and $j > 0$,

1. We know that $\widehat{x}_2 \cdot \widehat{Q}_2 \cdot \widehat{x}_2$ does not include x and is included in a unit cube (if $n = 6^+, 18, 26$) or a unit square (if $n = 6$) in $U_{26}^1(x, X)$. We also know that $\widehat{x}_2 - U_n - x$ and $\widehat{x}_2 - U_n - x$. From the definition of geodesic neighborhood, we must have $\widehat{Q}_2 \subset U_n^k(x, X)$, and thus $\widehat{x}_2 \cdot \widehat{Q}_2 \cdot \widehat{x}_2 \subset U_n^k(x, X)$. It then follows that $\widehat{x}_2$ and $\widehat{x}_2$ belong to the same component in $U_n^k(x, X)$. Hence, the contribution of $x\widehat{x}_2$ and $\widehat{x}_2 x$ in $\#(\gamma_l, \widehat{I}_1)$ is either 0 (if the component is not $\widehat{I}_1$) or 2 (otherwise). An example is shown by Fig. 17 in which $n = 6^+$, and we have $\widehat{Q}_2 = abc$. By extension, it can be seen that the contribution of $x\widehat{x}_2$ and $\widehat{x}_2 x, \dots, x\widehat{x}_i$ and $\widehat{x}_i x$ in $\#(\gamma_l, \widehat{I}_1)$ is an even number and, in a similar way, the contribution of $x\widehat{y}_2$ and $\widehat{y}_2 x, \dots, x\widehat{y}_i$ and $\widehat{y}_i x$ in $\#(\gamma_{l-1}, \widehat{I}_1)$ is an even number.
2. If $\widehat{u}_1 \neq x$, then the path $\widehat{x}_1 \cdot \widehat{Q}_1^{-1} \cdot \widehat{x}_1 \cdot \widehat{u}_1 \cdot \widehat{y}_1 \cdot \widehat{R}_1 \cdot \widehat{y}_1$ does not include x and is included in a unit cube (if $n = 6^+, 18, 26$) or a unit square (if $n = 6$) in $U_{26}^1(x, X)$. (Note that $\widehat{Q}_1^{-1}$ denotes the inverse path of $\widehat{Q}_1$, i.e. if $\widehat{Q}_1 = q_1 q_2 \cdots q_m$, then $\widehat{Q}_1^{-1} = q_m q_{m-1} \cdots q_1$.) We also know that $\widehat{x}_1 - U_n - x$ and $\widehat{y}_1 - U_n - x$. From the definition of geodesic neighborhood, we must have $\widehat{x}_1 \cdot \widehat{Q}_1^{-1} \cdot \widehat{x}_1 \cdot \widehat{u}_1 \cdot \widehat{y}_1 \cdot \widehat{R}_1 \cdot \widehat{y}_1 \subset U_n^k(x, X)$. Then it follows that $\widehat{x}_1$ and $\widehat{y}_1$

belong to the same component in $U_n^k(x, X)$. Hence the contribution of $\widehat{x}_1 x$ in $\#(\gamma_l, \widehat{I}_1)$ is the same as the contribution of $\widehat{y}_1 x$ in $\#(\gamma_{l-1}, \widehat{I}_1)$. Fig. 17 shows an example, where $\widehat{y}_1 = \widehat{y}'_1$, $\widehat{R}_1 = \emptyset$, $\widehat{Q}_1 = d$.

3. If $\widehat{u}_1 = x$, the discussion for $\widehat{x}_1 \cdot \widehat{Q}_1 \cdot \widehat{x}'_1$ is similar to case (1). We must have $\widehat{x}_1$ and $\widehat{x}'_1$ belong to the same component in $U_n^k(x, X)$. This is also true for $\widehat{y}_1$ and $\widehat{y}'_1$.
 4. The discussion for the contribution of $x\widehat{x}_{i+1}$ and $x\widehat{y}_{j+1}$ is similar to case (2) and (3). From (1)-(4) we conclude that $\#(\gamma_l, \widehat{I}_1)$ can be obtained from $\#(\gamma_{l-1}, \widehat{I}_1)$ by adding and subtracting even numbers, hence $\#(\gamma, \widehat{I}_1)$ is an odd number.
- b. If $i = 0$ or $j = 0$, the discussion is also similar to the above cases. For example, if $i = 0, j > 0$, and $\widehat{u}_1 \neq x$, it can be shown that $\widehat{y}_1$ and $\widehat{y}'_{j+1}$ belong to the same component in $U_n^k(x, X)$, and therefore $\#(\gamma_l, \widehat{I}_1) = \#(\gamma_{l-1}, \widehat{I}_1)$ is odd.

References

1. Ayala R, Dominguez E, Frances A, Quintero A. Homotopy in digital spaces. *Discrete Applied Mathematics* 2003;125:3–24.
2. Bai, Y.; Han, X.; Prince, JL. IEEE Workshop on Mathematical Methods in Biomedical Image Analysis. New York: Jun. 2006 Octree-based topology-preserving isosurface simplification..
3. Bai, Y.; Han, X.; Prince, JL. Information Processing in Medical Imaging. 2007. Octree grid topology preserving geometric deformable models for 3D medical image segmentation..
4. Bazin P, Pham DL. Topology-preserving tissue classification of magnetic resonance brain images. *IEEE Transactions on Medical Imaging* 2007;26(4):487–496. [PubMed: 17427736]
5. Bazin, P.; Ellingsen, L.; Pham, DL. Information Processing in Medical Imaging. 2007. Digital homeomorphisms in deformable registration.; p. 211-222.
6. Bertrand G. A new characterization of three-dimensional simple points. *Pattern Recognition Letters* 1994;15:169–175.
7. Bertrand G. Simple points, topological numbers and geodesic neighborhoods in cubic grids. *Pattern Recognition Letters* 1994;15:1003–1011.
8. Bertrand G, Couprie M. A model for digital topology. *Discrete Geometry for Computer Imagery* 1999;1568:227–239.
9. Brimkov V, Klette R. Border and surface tracing - theoretical foundations. *IEEE Transactions on Pattern Analysis And Machine Intelligence*. to appear.
10. de Andrade, MC.; Bertrand, G.; de Albuquerque Araujo, A. Proceedings of SPIE:Nonlinear Image Processing VIII. 1997. Segmentation of microscopic images by flooding simulation: a catchment-basins merging algorithm..
11. Droske, M.; Meyer, B.; Schaller, C.; Rumpf, M. An adaptive level set method for medical image segmentation.. In: Insana, MF.; Leahy, RM., editors. *Information Processing in Medical Imaging*. Springer Verlag; 2001. p. 416-422. LNCS 2082
12. Fourey S, Malgouyres R. A concise characterization of 3D simple points. *Discrete Applied Mathematics* 2005;125:59 C 80.
13. Frisken, SF.; Perry, RN.; Rockwood, AP.; Jones, TR. SIGGRAH. 2000. Adaptively sampled distance fields: A general representation of shape for computer graphics.; p. 249-254.
14. Han X, Xu C, Braga-Neto U, Prince JL. Topology correction in brain cortex segmentation using a multiscale graph-based approach. *IEEE Transactions on Medical Imaging* 2002;21:109–121. [PubMed: 11929099]

15. Han X, Xu C, Prince JL. A topology preserving level set method for geometric deformable models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2003;25:755–768.
16. Hatcher, A. *Algebraic Topology*. Cambridge University Press; Cambridge, UK: 2002.
17. Herman, GT., editor. *Geometry of digital spaces*. Birkhauser; 1998.
18. Ju T, Baker ML, Chiu W. Computing a family of skeletons of volumetric models for shape description. *Computer Aided Design* May;2007 39(5):352–360. [PubMed: 18449328]
19. Kenmochi, Y.; Kotani, K.; Imiya, A. *International Conference on Image Processing*. 1999. Marching cubes method with connectivity..
20. Khalimsky E, Kopperman R, Meyer PR. Computer graphics and connected topologies on finite ordered sets. *Topolog and its applications* 1990;36:1–17.
21. Kong TY. A digital fundamental group. *Computer Graphics* 1989;13:159–166.
22. Kong TY. On topology preservation in 2-D and 3-D thinning. *International Journal of Pattern Recognition and Artificial Intelligence* 1995;9:813–844.
23. Kong, TY. *Foundation of Image Understanding*. Kluwer; 2001. p. 73-93.chapter Digital topology
24. Kong TY. Topological adjacency relations on z^n . *Theoretical Computer Science* 2002;283:3–28.
25. Kong TY, Rosenfeld A. Digital topology: Introduction and survey. *CVGIP: Image Understanding* 1989;48:357–393.
26. Kong TY, Rosenfeld A. If we use 4- or 8-connectedness for both the objects and the background, the euler characteristics is not locally computable. *Pattern Recognition Letters* 1990;11:231–232.
27. Kong TY, Udupa JK. A justification of a fast surface tracking algorithm. *CVGIP: Graphical Model and Image Processing* 1992;54:162–170.
28. Kovalevsky V. Axiomatic digital topology. *Journal of Mathematical Imaging and Vision* 2006;26:41–58.
29. Lachaud J. Topologically defined iso-surfaces. *Lecture Notes in Computer Science* 1996;1176:245–256.
30. Lachaud J-O, Montanvert A. Continuous analogs of digital boundaries: A topological approach to isosurfaces. *Graphical Models* 2000;62:129–164.
31. Ma CM, Sonka M. A fully parallel 3D thinning algorithm and its applications. *Computer Vision And Image Understanding* 1996;64:420C433.
32. Malandain G, Bertrand G, Ayache N. Topological segmentation of discrete surfaces. *International Journal of Computer Vision* 1993;10(2):183–197.
33. Mangin J, Frouin V, Bloch I, Regis J, Lopez-Krahe J. From 3D magnetic resonance images to structural representations of the cortex topography using topology preserving deformations. *Journal of Mathematical Imaging and Vision* 1995;5:297–318.
34. Milne, RB. PhD thesis. Dept. Math., UC Berkely; 1995. Adaptive Level Sets Methods Interfaces..
35. Morgenthaler, DG. Technical report. Computer Vision Lab., Univ. of Maryland; 1981. Three-dimensional simple points: Serial erosion, parallel thinning and skeletonization..
36. Natarajan B. On generating topologically consistent isosurfaces from uniform samples. *Visual Computer* 1994;11(1):52–62.
37. Nielson, GM.; Hamann, B. *IEEE Visualization*. Los Alamitos; Calif.: 1991. The asymptotic decider: Resolving the ambiguity in marching cubes.; p. 83-91.
38. Niethammer M, Kalies WD, Mischaikow K, Tannenbaum A. On the detection of simple points in higher dimensions using cubical homology. *IEEE Transactions on Image Processing* 2006;15:2462–2469. [PubMed: 16900699]
39. Pudney C. Distance-ordered homotopic thinning: A skeletonization algorithm for 3d digital images. *Computer Vision and Image Understanding* 1998;72:404–413.
40. Rosenfeld A. Connectivity in digital pictures. *Journal of the Association for Computing Machinery* 1970;17:146–160.
41. Rosenfeld A. Arcs and curves in digital pictures. *Journal of the Association for Computing Machinery* 1973;20:81–87.
42. Rosenfeld, A.; Kak, AC. *Digital Picture Processing*. Academic Press, Inc.; 1982.

43. Saha P, Chaudhuri B, Majumder D. A new shape preserving parallel thinning algorithm for 3D digital images. *Pattern Recognition* 1997;30:1939C1955.
44. Saha PK, Chaudhuri BB. Detection of 3D simple points for topology preserving transformation with application to thinning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1994;16:1028–1032.
45. Saha PK, Chaudhuri BB. 3D digital topology under binary transformation with applications. *Computer Vision And Image Understanding* 1996;63:418–429.
46. Segonne, F.; Pons, J-P.; Grimson, E.; Fischl, B. *Computer Vision for Biomedical Image Applications*. 2005. Active contours under topology control genus preserving level sets.; p. 135-145.
47. Sethian JA. A fast marching level set method for monotonically advancing fronts. *Proc. Nat. Acad. Sci* 1996;93:1591–1595. [PubMed: 11607632]
48. Stelldinger P, Latecki L, Siqueira M. Topological equivalence between a 3D object and the reconstruction of its digital image. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2007;29(1):126–140. [PubMed: 17108388]
49. Teo P, Sapiro G, Wandell B. Creating connected representations of cortical gray matter for functional MRI visualization. *IEEE Transactions on Medical Imaging* 1997;16:852–863. [PubMed: 9533585]
50. Tsitsiklis JN. Efficient algorithm for globally optimal trajectories. *IEEE Trans. Automat. Contr* 1995;40(9):1528–1538.
51. Udupa JK. Multidimensional digital boundaries. *CVGIP:Graphical Models and Image Processing* 1994;56:311–323.
52. Xie W, Thompson RP, Perucchio R. A topology-preserving parallel 3d thinning algorithm for extracting the curve skeleton. *Pattern Recognition* 2003;36:1529–1544.

Figure 1.

(a) A 2D object with three “fingers”. (b) Coarse sampling grid cannot represent the true topology of the underlying object (assume 4-adjacency for the object) and key geometrical features are also lost. (c) An adaptive quadtree grid recovers the correct topology and geometrical features without increasing the sampling rate overall. In (b) and (c), the pink contour is a linear approximation of the boundary of the digital object defined on each grid.

Figure 2.

(a) The gray circle is the *root point*; the black squares denote the 6-neighbors; the white squares denote the 18-neighbors; and the gray squares denote the 26-neighbors. In (b)-(d), it is assumed that $n = 6$ for black points and $\bar{n} = 26$ for white points. Then the gray point is (b) a simple point, $T_6 = T_{26} = 1$; (c) a non-simple point, $T_6 = 2$, $T_{26} = 1$; and (d) a non-simple point, $T_6 = 0$, $T_{26} = 1$.

Figure 3.

Octree grid structure: Left panel shows the octree grid cells. Starting from root cell at level 0. $\{x_i, i = 1 \cdots 8\}$ are the eight vertices of the root cell. Right panel shows the tree hierarchy corresponding to the left panel: each square represents a cell on the octree. The white cells are leaf cells; the black cells are non-leaf cells.

Figure 4.

3D neighborhoods on an octree grid. In each example, the gray circle is the root point; the black squares denote the *E*-neighbors; the white squares denote the *S*-neighbors; and the gray squares denote the *C*-neighbors. (a)-(c) are single-level neighborhoods; (d) is a multi-level neighborhood.

Figure 5.

(a) The polygon $x_1x_3x_5x_7$ is a transition face, consisting of four finer resolution faces $x_1x_2x_9x_8$, $x_2x_3x_4x_9$, $x_8x_9x_6x_7$, and $x_9x_4x_5x_6$. The line segments x_1x_3 , x_3x_5 , x_5x_7 and x_7x_1 are four transition edges, each consisting of two finer edges. The line segments x_1x_5 and x_3x_7 are two transition face diagonals. (b) If we use E -adjacency for the white points, then all the points in the simple closed E -path $\gamma_y = y_1y_2y_3y_4y_5y_6y_7y_8y_1$ have single-level neighborhoods, and therefore each point is E -adjacent to only its successor and its predecessor. In the simple closed E -path $\gamma_x = x_1x_2x_3x_4x_5x_6x_7x_8x_1$, there are four points — x_1, x_3, x_5 , and x_7 — that have multi-level neighborhoods, and each of them is E -adjacent to two successors and two predecessors.

Figure 6.

Examples of invalid cases on transition edge/face. Assume S -adjacency for black points (X)

and E^+ -adjacency for white points ($\bar{X}$): (a) On this transition edge, $\#C_{\alpha}^C(x, \bar{X})=1$,

$\#C_{\alpha}^F(x, \bar{X})=2$; (b) On this transition edge, $\#C_{\alpha}^C(x, X)=1$, $\#C_{\alpha}^F(x, X)=2$; (c) On this transition

face, $\#C_{\alpha}^C(x, \bar{X})=0$, $\#C_{\alpha}^F(x, \bar{X})=1$; (d) On this transition face, $\#C_{\alpha}^C(x, X)=0$, $\#C_{\alpha}^F(x, X)=1$; and

(e-f) On this transition face, $\#C_{\alpha}^C(x, X)=1$, $\#C_{\alpha}^F(x, X)=2$.

Figure 7.
Demonstration of deformations: see text for details.

Figure 8.
Detection of handles through homotopic deformations on VOGs (see text for details).

Figure 9.

Illustration of geodesic neighborhoods: assume E^+ -adjacency for black points (X) and S -adjacency for white points ($\bar{X}$). (a) $N_C(x)$, where x is the gray point in the center. There are neighbor cells at three resolution levels, and suppose the levels are 1(largest) to 3(smallest).

(b) $N_E^3(x, X)$ (c) $N_S^2(x, \bar{X})$ (d) $U_6^3(x, X)$ (e) $U_{18}^2(x, \bar{X})$. After counting the number of connected components in (d) and (e), we get $T_{E^+}(x, X) = 2$ and $T_S(x, \bar{X}) = 1$.

Figure 10.

Choice of order of geodesic neighborhood for adjacency E^+ : (a) $N_C(x)$: a closed path that defines a handle is highlighted; (b) $N_E^4(x, X)$: the circled point is a 4th order neighbor point that causes the handle to exist in $N_E^4(x, X)$. (c) $N_C(x)$; (d) $N_E^2(x, X)$: there are two connected components in $N_E^2(x, X)$.

Figure 11.
Conversion of geodesic neighborhood into MLA form (see text for details).

Figure 12.
Equivalence of a deformation in $\tilde{N}_\alpha^k(x, X)$ and a deformation in $\tilde{U}_n^k(x, X)$.

Figure 13.

Skeleton computation of a chair phantom: (a) boundary surface of a chair phantom defined on a uniform grid; (b) a cross section view of the skeleton measure; (c) a cross section view of the resulting VOG overlaid on the skeleton measure; (d) boundary surface of the chair phantom defined on the VOG; (e) boundary surface of the skeleton computed on the uniform grid; (f) boundary surface of the skeleton computed on the VOG.

Figure 14.
Illustration for Appendix A (see text for details)

Figure 15.

Illustration for Appendix B. In each figure, the gray point is the root point x . All the white points belong to $\bar{x}$. All the other points not denoted in the neighborhood can either belong to X or $\bar{X}$. (a) $(\alpha, \bar{\alpha}) = (E, C)$ or (E^+, S) , then x is an E -isolated point. (b) $(\alpha, \bar{\alpha}) = (S, E^+)$, then x is an S -isolated point. (c) $(\alpha, \bar{\alpha}) = (C, E)$, then x is a C -isolated point.

Figure 16.
Illustration for Appendix C1 (see text for details).

Figure 17.
Illustration for Appendix C2 (see text for details).

Algorithm 1Check the validness constraint for the removal of a point x from X

INPUT: $N_C(x)$ **OUTPUT:** true/false**for** each transition edge/face inside $N_C(x)$ **do** Compute $\#C_a(I, X)$, $\#C_a(I', X)$, $\#C_{\bar{a}}(I, \bar{X})$ and $\#C_{\bar{a}}(I', \bar{X})$ **if** $\#C_a(I, X) \neq \#C_a(I', X)$ or $\#C_{\bar{a}}(I, \bar{X}) \neq \#C_{\bar{a}}(I', \bar{X})$ **then** **return** false /* removal of x causes an invalid case */ **end if** **end for** **return** true /* removal of x is valid */

Algorithm 2Construction of the geodesic neighborhood $U_6^3(x, X)$

INPUT: $N_C(x)$ **OUTPUT :** $U_6^3(x, X)$ Initialize U to be a $3 \times 3 \times 3$ array with all entries to be OFF**for** each neighbor cell C_i of x **do** tag the corresponding cube in U with the resolution level of C_i . **for** each E -neighbor of x in C_i **do** **if** this neighbor belongs to X **then** set the corresponding entry in U to be ON and tag the entry with the resolution level of C_i . **end if** **end for** **for** each S -neighbor of x in C_i **do** **if** this neighbor belongs to X and at least one of the adjacent E -neighbors also belong to X **then** set the corresponding entry in U to be ON and tag the entry with the resolution level of C_i . **end if** **end for** **for** each C -neighbor of x in C_i **do** **if** this neighbor belongs to X and at least one of the adjacent S -neighbors also belong to X **then** set the corresponding entry in U to be ON and tag the entry with the resolution level of C_i . **end if** **end for****end for**

Algorithm 3

Counting n -components in $U_n^k(x, X)$

Procedure *CountCCn*(U)

INPUT : $U_n^k(x, X)$

OUTPUT: $T_n(x, X)$

initialize queue Q

set counter $t = 0$

for each n -adjacent direction **do**

 find the n -neighbor p of x in that direction

if $U(p) = \text{ON}$ **then**

if p, x and a cube they share are all tagged with a same resolution level **then**

$t++$

 set $U(p) = \text{OFF}$

 push point p into Q

end if

end if

while Q is not empty **do**

 pop point q from Q

DepthFirstTraversal(U, q)

end while

end for

return t ;

Procedure *DepthFirstTraversal*(U, q)

INPUT : $U_n^k(x, X)$, point q

for each n -adjacent direction **do**

 find q 's n -neighbor r in that direction

if $U(r) = \text{ON}$ **then**

if r, q and a cube they share are all tagged with a same resolution level **then**

 set $U(r) = \text{OFF}$

 push point r into Q

end if

end if

end for