# Theory and Algorithms for Constructing Discrete Morse Complexes from Grayscale Digital Images

Vanessa Robins, Peter John Wood, and Adrian P. Sheppard

**Abstract**—We present an algorithm for determining the Morse complex of a two or three-dimensional grayscale digital image. Each cell in the Morse complex corresponds to a topological change in the level sets (i.e., a critical point) of the grayscale image. Since more than one critical point may be associated with a single image voxel, we model digital images by cubical complexes. A new homotopic algorithm is used to construct a discrete Morse function on the cubical complex that agrees with the digital image and has exactly the number and type of critical cells necessary to characterize the topological changes in the level sets. We make use of discrete Morse theory and simple homotopy theory to prove correctness of this algorithm. The resulting Morse complex is considerably simpler than the cubical complex originally used to represent the image and may be used to compute persistent homology.

**Index Terms**—Discrete Morse theory, computational topology, persistent homology, digital topology.

## 1 Introduction

The rise of three-dimensional (3D) imaging technology has sharpened the need for quantitative analysis of large 3D images. Typically, these quantities are geometric in nature—for example, volume, surface area, and curvature. Topological information is also useful as it is independent of geometric measures, but there has been limited success to date in adapting the tools of computational topology to 3D digital images. These tools require a combinatorial complex that encodes the topological structure of an object or function. We present here a fast and robust algorithm for deriving such a complex from grayscale images, and prove that this complex accurately reflects the topology of the level sets of the image function.

A grayscale digital image is a function defined on a discrete lattice so the first issue is how to endow this space with a topology. As Kovalevsky [1] shows, a cubical complex is the only topologically consistent model of a digital image. Important work has also been done in this area by Couprie et al. [2], [3], [4] who investigate in detail the properties of two, three, and four-dimensional digital images as cubical complexes. We treat voxels as vertices in the cubical complex; the above authors treat voxels as cubes. These two approaches are dual and carry the same topological information.

- *V. Robins and A.P. Sheppard are with the Department of Applied Mathematics, Research School of Physics and Engineering, The Australian National University, Canberra ACT 0200, Australia. E-mail: {Vanessa.Robins, Adrian.Sheppard}@anu.edu.au.*
- *P.J. Wood is with the Resource Management in Asia-Pacific Program, College of Asia and the Pacific, The Australian National University, Canberra ACT 0200, Australia. E-mail: peter.j.wood@anu.edu.au.*

*Manuscript received 26 Mar. 2009; revised 18 Oct. 2009; accepted 21 Jan. 2010; published online 5 May 2011.*
*Recommended for acceptance by R. Ramamoorthi.*
*For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2009-03-0192.*
*Digital Object Identifier no. 10.1109/TPAMI.2011.95.*

The cubical complex is a valid foundation for the algorithms from computational topology that we wish to apply, such as persistent homology [5]. However, the runtimes for these algorithms can scale as $\mathcal{O}(n^3)$, where $n$ is the number of cells in the complex, so it is impractical to apply them directly to large 3D images.[1] The algorithms presented in this paper construct a simplified complex that represents the same topological information as the cubical complex. Earlier tools for summarizing structure in grayscale images are the watershed transform [7] and the component tree [8]. Both of these methods proceed by studying changes in the connected components of the level sets (isosurfaces) of a grayscale function. The complex we build is a generalization of the watershed and component tree, called the *Morse complex*, that encodes all topological changes in the level sets of a function.

The Morse complex has its origins in the work of Cayley [9] and Maxwell [10], who described the partition of a landscape into hills or dales, and was later more generally defined within Morse theory [11]. Morse theory relates the critical points of a smooth real-valued function, $f$, to the topology of the manifold on which $f$ is defined by studying the negative *gradient flow* of $f$ (the negative gradient is used because the gradient points uphill, but physical flows move downhill). Cells in the Morse complex are defined by regions of uniform flow behavior as follows: Imagine water droplets flowing over a smooth height landscape: Droplets that flow to the same minimum define 2D regions (dales or basins); boundaries between basins are defined by ridge lines passing through saddles; maxima are isolated points where several basins touch. What we have just described are the *stable manifolds* of the critical points. Conversely, we can consider regions whose flow lines have the same origin—these are the *unstable manifolds*. For certain smooth

1. We note that there has been some success in adapting 2D persistence computation to 3D problems [6].

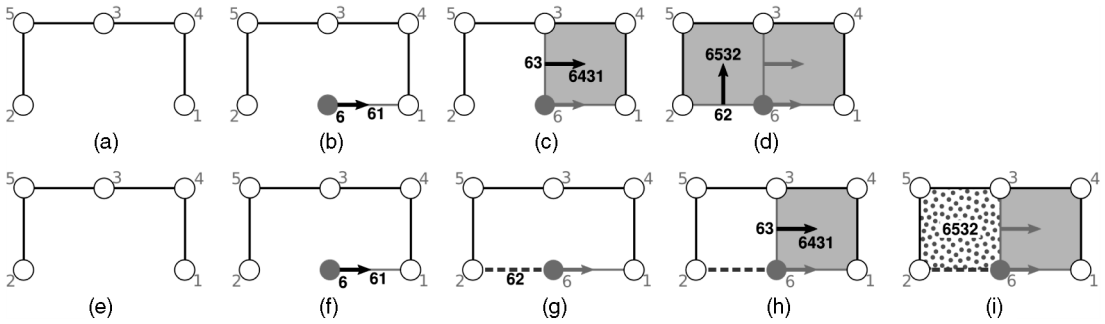


Fig. 1. Processing the lower star of vertex 6 (a subset of a 2D digital image). Pixels are labeled by their value $g(x)$ and other cells by the word defined by $G(\alpha)$. Using this labeling, the lower star contains the cells 6, 61, 62, 63, 6431, and 6532 listed in lexicographic order. (a)-(d) show the operation of our homotopic algorithm *ProcessLowerStars*, while (e)-(i) show the action of a nonhomotopic algorithm. Steps (a) and (e) show the *lower link* of vertex 6; these are the cells of $\mathcal{K}_5$ that are faces of cells in $L(6)$. Algorithm 1 proceeds as follows: (b) Vertex 6 is paired with the edge 61. (c) Edge 63 pairs with face 6431 (face 6532 is not yet in *PQone*). (d) Edge 62 pairs with face 6532. A nonhomotopic algorithm may introduce spurious critical cells (62 and 6532) if the lower star is traversed with the lexicographic ordering. (f) Vertex 6 is paired with the edge 61. (g) Edge 62 is inserted as a critical cell (dashed line). (h) Edge 63 pairs with face 6431. (i) Face 6532 is inserted as a critical cell (spots).

functions we can also take the intersection of stable and unstable manifolds; the resulting partition is called the *Morse-Smale complex* [12].

Several issues must be resolved when translating the above picture to a computational setting: adapting Morse theory to nonsmooth functions and manifolds, defining critical points, approximating the gradient flow, devising efficient algorithms. An extensive review of various approaches to solving these problems may be found in [13]. Our approach is based on Forman's *discrete Morse theory* [14], [15]. Forman's theory is an elegant adaption of classical Morse theory to certain functions defined on cell complexes. Gradient arrows are given by a one-to-one pairing of cells with faces; unpaired cells are critical points. The flow follows gradient arrows through cells of each dimension; the Morse complex arises naturally in this setting from the flow paths between critical cells.

The computational challenge is to extend the input—a grayscale digital image which is defined only on the vertices of a cubical complex—to a valid Morse function defined on every cell of the complex. Our solution to this problem provides the first algorithm that is provably correct in the sense that the critical cells of the discrete Morse function are exactly the number and type needed to reflect changes in topology of the lower level cuts of the grayscale function. Our result is stated in Theorem 11 and proven in Section 4.

The main influences on our work are discussed below. First, our use of cubical complexes and simple homotopy was inspired by Couprie and Bertrand's [4] characterization of 2D, 3D, and 4D simple voxels. Edelsbrunner et al. [12] introduced 2D Morse-Smale complexes as a tool in computer graphics and computational topology. Their approach, based on Banchoff's extension of Morse theory to piecewise-linear manifolds and functions [16], is most effective for functions defined on the vertices of triangulated surfaces. Adaptions of their algorithms to 3D simplicial complexes ran into a number of technical difficulties in practice [17], [18].

Recently, several authors have proposed algorithms for building a discrete Morse function from a scalar function defined on the vertices of a mesh. Lewiner et al. [19], [20], devised an algorithm to build an optimal discrete Morse function on a mesh without an initial scalar function. Later, his thesis [21] presented a method for building a discrete Morse function to match a scalar function defined on the mesh vertices. This approach requires extra pre and postprocessing steps to avoid inconsistencies between the critical cells of the discrete Morse function and the original scalar function. King et al. [22] also develop a multistage algorithm to build a discrete Morse function from a function defined on the vertices of a simplicial complex. They prove limited results about the correctness of critical cells identified by their algorithms. Gyulassy et al. [23] apply discrete Morse theory to grayscale image data. This algorithm for computing the discrete gradient, however, can introduce extra critical cells which may be canceled later (see Fig. 1 in Section 3). This makes correctness proofs difficult—the authors of [23] make no formal statements about the relationship between the discrete gradient and the scalar field it is derived from. Improvements are discussed in Gyulassy's thesis [24]; in particular he is able to treat flat regions without introducing spurious critical cells.

The remainder of this paper is organized as follows: Section 2 covers the relevant background material: the representation of grayscale images as cubical complexes, basic definitions from homology and CW complexes, simple homotopy theory, and discrete Morse theory. Section 3 presents our homotopic algorithm for building a discrete Morse function from a digital image and contains a discussion of the computational complexity of our approach. Section 4 contains the proof of Theorem 11 that each cell of the Morse complex corresponds to a change in topology between successive lower level sets of the cubical complex. We then apply our algorithm to some 2D and 3D images and use the resulting Morse complex as the basis for persistent homology computations; results are presented in Section 5.

## 2 Background

### 2.1 Digital Images as Cubical Complexes

This paper focuses on 3D grayscale digital images. Such an image is a function $g : D \to \mathbb{R}$, where $D \subset \mathbb{Z}^3$ is typically a rectangular subset of the discrete lattice

$$D = \{(i, j, k) \mid 0 \le i \le I,\ 0 \le j \le J,\ 0 \le k \le K\}.$$

A point $x \in D$ is called a voxel (or pixel in two dimensions). The algorithms presented in Section 3 apply without adjustment to 2D digital images.

We model digital images by cubical complexes. In our setting, the voxels $(i, j, k) \in D$ are the vertices (0-cells) of the complex. Higher dimensional cells are the unit edges (1-cells) between voxels whose coordinates differ by one in a single axis, unit squares (2-cells), and unit cubes (3-cells). We write $\mathcal{K}(D)$ (or simply $\mathcal{K}$) for the collection of all such $p$-cells built from voxels in $D$ with $p = 0, 1, 2, 3$. We often denote the dimension of a cell by a superscript: $\alpha^{(p)}$. A cell $\alpha^{(p)} \in \mathcal{K}$ is a *face* of another cell $\beta^{(q)}$ if $p < q$ and the vertices of $\alpha$ are a subset of the vertices of $\beta$. If this is the case, then we write $\alpha^{(p)} < \beta^{(q)}$. We can also say that $\beta$ is a *coface* of $\alpha$. A set of cells $S \subset \mathcal{K}$ is a *complex* if for any $\alpha \in S$, all its faces are also in $S$.

Now consider the set of all voxels with grayscale values less than a threshold $t$; this is called a *lower level cut*:

$$D_t = \{x \in D \mid g(x) \leq t\}.$$

In fact, we will mostly consider level cuts on the cubical complex, where a cell $\alpha$ is included if all its vertices $x \in \alpha$ are below the threshold

$$\mathcal{K}_t = \{\alpha \in \mathcal{K} \mid g(x) \leq t,\ \forall x \in \alpha\}. \tag{1}$$

We note that the connectivity of level cuts of the cubical complex is equivalent to considering $D_t$ with the standard 6-neighborhood of 3D digital topology: Voxel $(i, j, k)$ is connected to a voxel on a face diagonal, say $(i + 1, j + 1, k)$, if and only if at least one of the voxels $(i + 1, j, k)$ and $(i, j + 1, k)$ are present.

The main goal of our work is to describe how the topology of the lower level cuts changes as the threshold is varied. For $t$ less than the minimal grayscale value, $\mathcal{K}_t$ is empty; as $t$ increases, cells are added so that for any $t_1 < t_2$, $\mathcal{K}_{t_1}$ is a subcomplex of $\mathcal{K}_{t_2}$. Eventually, when $t$ is larger than the maximal grayscale value, we obtain the entire domain of the image: $\mathcal{K}_t = \mathcal{K}$. Such a sequence of nested complexes is called a *filtration*. We quantify the changes in topology over this sequence of subcomplexes by computing the persistent homology of the filtration. Definitions of homology and persistence are reviewed in the following section.

### 2.2 CW Complexes and Homology

Homology theory uses algebraic groups to encode the topological structure of a complex. The complex can be simplicial (i.e., a triangulation), cubical (as above), a more general geometric object known as a *CW complex*, or even an abstract chain complex. Cells in a CW complex are no longer restricted to triangles or cubes, but are homeomorphic to euclidean balls. We make a formal definition of CW complex below, and define homology on CW complexes, as this is the most general setting we require. The reader is referred to [25] or [15, Section 1] for more detailed information on CW-complexes, and to [25], [26] for further background on homology theory.

A finite CW complex, $X$, is constructed inductively starting with $X_0$, a finite set of 0-cells (i.e., vertices), then attaching some number of 1-cells (edges) to the vertices in $X_0$ to obtain the 1-skeleton $X_1$, then attaching 2-cells to $X_1$, and so on up to some maximal dimension $n$

$$\emptyset \subset X_0 \subset X_1 \subset \ldots \subset X_n = X.$$

The *attaching map*, $\phi$, for a $p$-cell $\alpha^{(p)}$, is a continuous function from the boundary $\partial\alpha^{(p)}$ (which is a $(p-1)$-sphere) into the $(p-1)$-skeleton $X_{p-1}$, i.e., $\phi : S^{p-1} \to X_{p-1}$.

The $p$-skeletons and attaching maps define the geometry of the cell complex, and we now build an algebraic structure on top of this by defining $p$-chains as formal sums of $p$-cells. The first step is to choose a coefficient group. This is typically the integers, but we will work with $\mathbb{Z}_2$ (integers with addition modulo 2) as this means we do not need to keep track of the orientations of each cell. Let $X$ be a CW-complex, a *$p$-chain* of $X$ with $\mathbb{Z}_2$-coefficients is a function from the $p$-cells of $X$ to $\mathbb{Z}_2$ that vanishes on all but finitely many $p$-cells. We can add two $p$-chains by adding their values. This defines a group (the *chain group*) that we denote by $C_p(X, \mathbb{Z}_2)$.

Homology theory tells us that there exist boundary maps $\partial_p : C_p(X, \mathbb{Z}_2) \to C_{p-1}(X, \mathbb{Z}_2)$ so that $\partial_{p-1} \circ \partial_p = 0$. We can then define a *chain complex*

$$0 \to C_n(X, \mathbb{Z}_2) \xrightarrow{\partial_n} C_{n-1}(X, \mathbb{Z}_2) \xrightarrow{\partial_{n-1}} \cdots \xrightarrow{\partial_1} C_0(X, \mathbb{Z}_2) \to 0.$$

We define the *pth cycle group* $Z_p(X) = \mathrm{Ker}\partial_p$, and the *pth boundary group* $B_p(X) = \mathrm{Im}\partial_{p+1}$. Homology groups are the quotients

$$H_p(X, \partial) = \frac{\mathrm{Ker}\ \partial_p}{\mathrm{Im}\ \partial_{p+1}} = \frac{Z_p(X)}{B_p(X)}.$$

The elements of $H_p(X, \partial)$ are equivalence classes of $p$-cycles such that two cycles are equivalent if their difference is the boundary of a $(p+1)$-chain.

The basic topological structure of a space is quantified by the number of independent cycles in each homology group. Specifically, the rank of the $p$th homology group $H_p(X, \partial)$ is called the *pth Betti number*. Simply stated, for a 3D object the zeroth Betti number counts the number of connected components, the first Betti number counts the number of "handles," and the second Betti number counts the number of enclosed "voids."

We can examine the topology of a filtration by calculating the Betti numbers at each stage of the filtration, but this does not distinguish between cycles that exist in a filtration for a short amount of time, and cycles that persist in the filtration. To do this we make use of persistent homology, [5], [27].

Let $(X_l)_{l=0}^{L}$ be a filtration of CW-complexes. Consider the addition of a single $p$-cell $\alpha$ to the subcomplex $X_i$. The boundary of $\alpha$ must already be in $X_i$, so $\alpha$ either creates a new $p$-cycle or fills in a $(p-1)$-cycle. The idea behind persistent homology is that these events are paired and define the lifetime of a topological feature. Specifically, suppose that $z$ is a $p$-cycle created by adding the cell $\alpha$ at stage $i$ of the filtration and that $z'$ is a $p$-cycle that is homologous to $z$ with $z' = \partial\beta$, where $\beta$ is a $(p+1)$-cell added at stage $j$ of the filtration. The *persistence* of $z$, and of its homology class $[z]$, is equal to $j - i - 1$. We call $\alpha$ the *creator* and $\beta$ the *destroyer* of $[z]$. We also call a creator a *positive* cell, and a destroyer a *negative* cell. If a cycle class does not have a destroyer, then its persistence is infinite. The *$k$-persistent pth homology group* of $(X_l)_{l=0}^{L}$ consists of $p$-cycles in $X_l$ that are equivalent if their difference bounds in $X_{l+k}$:

$$H_p^{l,k} = Z_p(X_l)/(B_p(X_{l+k}) \cap Z_p(X_l)). \quad (2)$$

For some filtrations, we can define a function $F$ on the cells such that $F$ is nondecreasing with the filtration order. We can then look at persistence based on the values of $F$. This is known as *time-based persistence* [5].

### 2.3 Simple Homotopy and Discrete Morse Theory

Discrete Morse theory is a combinatorial analogue of Morse theory developed by Forman [14], [15]. Some concepts in discrete Morse theory have their origins in Whitehead's simple homotopy theory [28], so we introduce simple homotopy theory first. We also use simple homotopy theory to prove the validity of our algorithms in Section 4. The definitions we give below are for cubical complexes, but the concepts can be extended to general CW complexes [14].

For a cubical complex $\mathcal{K}$ with cell $\beta^{(p)}$, we say that $\beta$ has a *free face* $\alpha^{(p-1)}$ if $\alpha$ is a face of $\beta$ and $\alpha$ has no other cofaces. We call $(\alpha, \beta)$ a *free pair*. If we remove a free pair from $\mathcal{K}$, we obtain a subcomplex of $\mathcal{K}$ that is called an *elementary collapse* of $\mathcal{K}$. We say that a cubical complex $\mathcal{K}$ *collapses* to a subcomplex $\mathcal{K}'$ if there is a finite sequence of elementary collapses from $\mathcal{K}$ to $\mathcal{K}'$. If a cubical complex collapses to a point, it is *collapsible*. The inverse of a collapse is an *expansion*. We say that two cubical complexes are *simple homotopy equivalent* if there is a sequence of collapses and expansions from one complex to the other. An elementary collapse determines a strong deformation retraction. It therefore follows that if two cubical complexes are simple homotopy equivalent, then they are homotopy equivalent.

We now introduce discrete Morse functions. A function, $f : \mathcal{K} \to \mathbb{R}$ is a *discrete Morse function* if for every $\alpha^{(p)} \in \mathcal{K}$, $f$ takes a value less than or equal to $f(\alpha)$ on at most one coface of $\alpha$ and takes a value greater than or equal to $f(\alpha)$ on at most one face of $\alpha$. In other words,

$$\#\{\beta^{(p+1)} > \alpha \mid f(\beta) \leq f(\alpha)\} \leq 1 \quad (3)$$

and

$$\#\{\gamma^{(p-1)} < \alpha \mid f(\gamma) \geq f(\alpha)\} \leq 1, \quad (4)$$

where $\#$ denotes the number of elements in the set. A cell, $\alpha^{(p)}$, is *critical* if all cofaces take strictly greater values and all faces are strictly lower.

A cell $\alpha$ can fail to be critical in two possible ways. There can exist $\gamma < \alpha$ such that $f(\gamma) \geq f(\alpha)$ or there can exist $\beta > \alpha$ such that $f(\beta) \leq f(\alpha)$. Forman [14, Lemma 2.5] shows that these two possibilities are exclusive, they cannot be true simultaneously for a given cell $\alpha$. Thus, each noncritical cell $\alpha$ may be paired either with a noncritical cell that is a coface of $\alpha$, or with a noncritical cell that is a face of $\alpha$.

As noted by Forman [15, Section 3], it is not easy to construct discrete Morse functions on a given complex; it is usually simpler to work with a discrete vector field. A *discrete vector field*, $V$, is a collection of pairs $(\alpha^{(p)} < \beta^{(p+1)})$ of cells in $\mathcal{K}$ such that each cell is in at most one pair of $V$. A discrete Morse function defines a discrete vector field by pairing $\alpha^{(p)} < \beta^{(p+1)}$ whenever $f(\beta) \leq f(\alpha)$. The critical cells are precisely those that do not appear in any pair. Discrete vector fields that arise from Morse functions are called *gradient vector fields*.

It is natural to consider the flow associated with a vector field and in the discrete setting the analogy of a flow-line is a $V$-path. A *$V$-path* is a sequence of cells

$$\alpha_0^{(p)}, \beta_0^{(p+1)}, \alpha_1^{(p)}, \beta_1^{(p+1)}, \alpha_2^{(p)}, \ldots, \beta_{r-1}^{(p+1)}, \alpha_r^{(p)},$$

where $(\alpha_i, \beta_i) \in V$, $\beta_i > \alpha_{i+1}$, and $\alpha_i \neq \alpha_{i+1}$ for all $i = 0, \ldots, r-1$. A $V$-path is a *nontrival closed $V$-path* if $\alpha_r = \alpha_0$ for $r \geq 1$. Forman shows that a discrete vector field is the gradient vector field of a discrete Morse function if and only if there are no nontrivial closed $V$-paths ([14, Theorem 9.3]).

The connection between discrete Morse theory and simple homotopy arises from the pairs in the gradient vector field. For a cubical complex $\mathcal{K}$ with discrete Morse function $f$, and any real number $c$, define the *level subcomplex* $\mathcal{K}(c)$ by including all cells with $f(\alpha) \leq c$ and all their faces:

$$\mathcal{K}(c) = \bigcup_{f(\alpha) \leq c} \bigcup_{\gamma \leq \alpha} \gamma.$$

Notice that a sequence of subcomplexes $\mathcal{K}(c_i)$ for increasing values of $c_i$ defines a filtration. The following result [14, Theorem 3.3] relates this filtration to simple homotopy theory.

**Lemma 1.** *If there are no critical simplices $\alpha$ with $f(\alpha) \in (a, b]$, then $\mathcal{K}(b)$ collapses to $\mathcal{K}(a)$.*

The sequence of elementary collapses are defined by pairs $(\alpha^{(p)}, \beta^{(p+1)})$ from the gradient vector field.

Alternatively, if level subcomplexes $\mathcal{K}(a)$ and $\mathcal{K}(b)$ differ by a single critical $p$-cell, $\sigma^{(p)}$, then we can define an attaching map on the boundary of $\sigma$ (see [14, Theorem 3.4]). Thus, a discrete Morse function on a cubical complex induces a CW-complex [14, Corollary 3.5].

**Lemma 2.** *Let $\mathcal{K}$ be a cubical complex with discrete Morse function $f$. Then, $\mathcal{K}$ is homotopy equivalent to a CW-complex $X$ with exactly one cell of dimension $p$ for each critical cell in $\mathcal{K}$ of dimension $p$.*

We now define a discrete analogue of what is sometimes known as the "Thom-Smale-Witten" complex [29]. The result is an abstract chain complex that Forman calls the Morse complex, and that we call the *Morse chain complex*.

Let $C_p(\mathcal{K}, \mathbb{Z}_2)$ be the group of $p$-chains with $\mathbb{Z}_2$ coefficients (see Section 2.2). Let $\mathcal{M}_p \subseteq C_p(\mathcal{K}, \mathbb{Z}_2)$ be the subgroup generated by the critical $p$-cells. We define a boundary map $\partial_p : \mathcal{M}_p \to \mathcal{M}_{p-1}$ as follows: For any critical $p$-cell $\beta$ set

$$\partial\beta = \sum_{\alpha \in \mathcal{M}_{p-1}} c_{\alpha,\beta}\alpha. \quad (5)$$

The coefficients $c_{\alpha,\beta}$ encapsulate the induced orientations and number of $V$-paths between the maximal (i.e., co-dimension-1) faces of $\beta$ and the critical $(p-1)$-cell $\alpha$, and are defined as follows: Let $\Gamma(\beta, \alpha)$ be the set of all $V$-paths from maximal faces of $\beta$ to $\alpha$. Because we compute homology with $\mathbb{Z}_2$ coefficients, orientations are ignored and

$$c_{\alpha,\beta} = \begin{cases} 1 & \text{if } \#\Gamma(\beta, \alpha) \text{ is odd,} \\ 0 & \text{if } \#\Gamma(\beta, \alpha) \text{ is even.} \end{cases} \quad (6)$$

With the boundary operator so defined, we have the following result (see [14, Section 8], or [15, Section 7]).

**Theorem 3.** *Let $\mathcal{M}_p$ be the chain group of critical $p$-cells. The Morse chain complex*

$$0 \to \mathcal{M}_n \xrightarrow{\partial_n} \mathcal{M}_{n-1} \xrightarrow{\partial_{n-1}} \cdots \xrightarrow{\partial_0} \mathcal{M}_0 \to 0, \tag{7}$$

*calculates the homology of $\mathcal{K}$. That is, if we define*

$$H_p(\mathcal{M}, \partial) = \frac{\operatorname{Ker} \partial_p}{\operatorname{Im} \partial_{p+1}},$$

*then for each $p$,*

$$H_p(\mathcal{M}, \partial) \cong H_p(\mathcal{K}, \mathbb{Z}_2).$$

This theorem implies that the persistent homology of the digital image can be computed from the Morse chain complex.

## 3 The Algorithms

Our approach to characterizing the topology of the level cuts of a grayscale function proceeds in two stages. First, we construct a discrete vector field on the cubical cell complex using simple-homotopy expansions; we use this vector field to build the Morse chain complex by following $V$-paths.

### 3.1 Constructing the Discrete Vector Field

Recall from (1) that a filtration of the cubical complex $\mathcal{K}$ is the sequence of subcomplexes $\mathcal{K}_t$ such that $\mathcal{K}_t$ contains the cells in $\mathcal{K}$ that have no vertex with a grayscale value greater than $t$. The subcomplexes thus correspond to the lower level cuts of the digital image. The goal of our algorithm is to construct a discrete Morse function on the cubical complex whose critical cells exactly match the changes in topology between successive elements of this filtration. We achieve this by using simple homotopy expansions to grow from one subcomplex to the next, and introduce critical cells only when a simple homotopy pairing cannot be found.

It is convenient for our method to require that the value of $g$ on each voxel is distinct so that the voxels in $D$ can be ordered as

$$g(x_0) < g(x_1) < \cdots < g(x_N).$$

To ensure unique values, $g$ may need to be perturbed with a tie-breaking scheme. One way of doing this is to add a linear ramp to $g$ as follows: The image has a finite number of voxels, so there exists some $\eta > 0$ such that for any two voxels $x$ and $y$, either $g(x) = g(y)$ or $|g(x) - g(y)| > \eta$. Define a perturbed grayscale function $g'$ by

$$g'(i,j,k) = g(i,j,k) + \eta \frac{(i + Ij + IJk)}{3IJK}, \tag{8}$$

where $I, J, K$ are the dimensions of the image domain. The effect of this perturbation can be removed later by persistence canceling to level $\eta$.

When the $g$-values are unique, we know that the subcomplex $\mathcal{K}_{t_i}$ differs from its predecessor in the filtration $\mathcal{K}_{t_{i-1}}$ by the single voxel $x_i = g^{-1}(t_i)$ and the 1, 2, and 3-cells for which $x_i$ is the highest valued voxel. This set of cells in the neighborhood of $x_i$ is called the *lower star*. To be explicit the lower star, $L(x)$ is

$$L(x) = \{\alpha \in \mathcal{K} \mid x \in \alpha \text{ and } g(x) = \max_{y \in \alpha} g(y)\}.$$

The lower stars of all voxels $x \in D$ form a disjoint partition of $\mathcal{K}$.

We now present an algorithm, *ProcessLowerStars*, that, for each voxel $x_i$, finds a sequence for adding the cells from $L(x_i)$ to $\mathcal{K}_{t_{i-1}}$ to grow it into $\mathcal{K}_{t_i}$. To avoid spurious critical cells, our algorithm aims to leave a minimal number of cells unpaired. This is achieved by utilizing simple homotopic expansions where possible. Since pairing occurs only between cells in the same lower star, each lower star can be treated independently, in parallel if need be. Pseudocode for *ProcessLowerStars* is given in Algorithm 1.

**Algorithm 1.** ProcessLowerStars($D$, $g$)
**Input:** $D$ digital image voxels.
**Input:** $g$ grayscale values on voxels.
**Output:** $C$ critical cells.
**Output:** $V$ discrete vector field $V[\alpha^{(p)}] = \beta^{(p+1)}$.

```
 1: for x ∈ D do
 2:   if L(x) = {x} then {x is a local minimum}
 3:     add x to C
 4:   else
 5:     δ := the 1-cell in L(x) such that G(δ) is minimal
 6:     V[x] := δ
 7:     add all other 1-cells from L(x) to PQzero.
 8:     add all cells α ∈ L(x) to PQone such that α > δ
        and num_unpaired_faces(α) = 1
 9:     while PQone ≠ ∅ or PQzero ≠ ∅ do
10:       while PQone ≠ ∅ do
11:         α := PQone.pop_front
12:         if num_unpaired_faces(α) = 0 then
13:           add α to PQzero
14:         else
15:           V[pair(α)] := α
16:           remove pair(α) from PQzero
17:           add all cells β ∈ L(x) to PQone such that
              (β > α or β > pair(α)) and
              num_unpaired_faces(β) = 1
18:         end if
19:       end while
20:       if PQzero ≠ ∅ then
21:         γ := PQzero.pop_front
22:         add γ to C
23:         add all cells α ∈ L(x) to PQone such that
            α > γ and num_unpaired_faces(α) = 1
24:       end if
25:     end while
26:   end if
27: end for
```

*ProcessLowerStars* is based on two priority queues of cells, *PQone* and *PQzero*, that are described in greater detail below. The algorithm also requires a function that returns the cells in the lower star of a voxel, $L(x)$, and another function, *num_unpaired_faces*($\alpha$), which returns the number of faces of $\alpha$ that are in $L(x)$ and have not yet been inserted in either $C$ or $V$ (the lists of critical cells and discrete vector

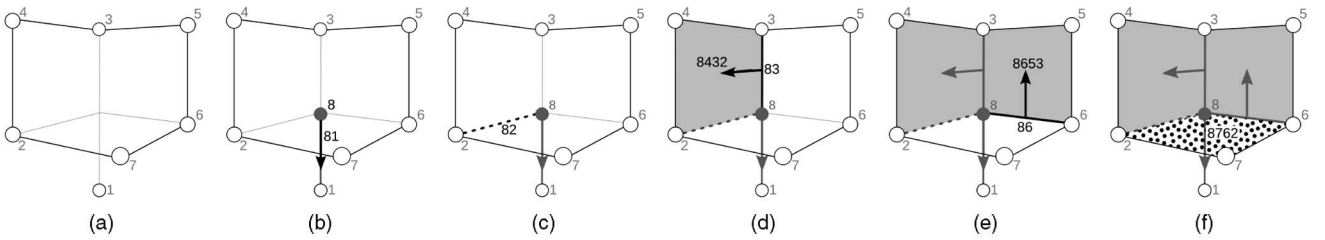


Fig. 2. In this example, we work through one iteration of the **for** loop in *ProcessLowerStars* using a voxel with grayscale value 8. We label voxels by their value $g(x)$ and other cells by the word defined by $G(\alpha)$. The lower star contains the cells 8, 81, 82, 83, 8432, 86, 8653, and 8762 listed in lexicographic order. In (a) we show the *lower link* of vertex 8 as black edges; these are the faces of cells in $L(8)$ that belong to the complex $\mathcal{K}_7$. The pale edges are included to assist visualization. The Algorithm 1 proceeds as follows:

- Vertex 8 is paired with the edge 81 (b).
- Edges 82, 83, and 86 are added to the priority queue *PQzero*. There are no cells adjacent to 81 with a single unpaired face so *PQone* is empty. Edge 82 is popped from *PQzero*; it is critical and marked as a dashed line (c).
- Edge 82 has two cofaces 8432 and 8762 that are added to *PQone*. Square 8432 is popped from *PQone*; it is paired with its free face, edge 83 (d).
- Square 8653 is a coface of edge 83 and added to *PQone*. Square 8653 is then popped from *PQone*, and paired with edge 86 (e).
- Square 8762 is popped from *PQone*; all of its faces have now been paired, so 8762 is added to *PQzero*. Finally, square 8762 is popped from *PQzero*; it is a critical 2-cell marked with spots (f).

The algorithm has reordered the cells of $L(8)$ as 81, 8, **82**, 8432, 83, 8653, 86, and **8762** (the critical cells are boldface).

field pairs). When there is a single available unpaired face for the cell $\alpha$, we call this face $pair(\alpha)$. The priority queues require an ordering, $G(\alpha)$, of the cells in each lower star. Any ordering based primarily on the maximal $g$ value of the vertices of the cell and in which a cell is ranked after its faces will suffice. Indeed, the results in Section 4 show that for 2D and 3D complexes the number and type of critical cells found by *ProcessLowerStars* are independent of this ordering—a property of our algorithm that arises from the homotopic growth of each lower star. The ordering we implement is based on lexicographic ordering of sequences defined by listing the $g$-values from the vertices of a cell in decreasing order. Specifically, given a cell, $\alpha \in L(x)$, with vertices $\{x, y_1, \ldots, y_k\}$, define

$$G(\alpha) = (g(x), g(y_{i_1}), \ldots, g(y_{i_k})),$$
$$\text{where } g(x) > g(y_{i_1}) > \cdots > g(y_{i_k}).$$

The first cell considered in each step of the main loop in *ProcessLowerStars* is the 0-cell $x$ itself. If $L(x)$ is empty, $x$ is a local minimum and is critical (line 2). Otherwise, $x$ is paired with its lowest incident edge (line 6). The algorithm then constructs the lower star via cell pairs that are simple homotopy expansions. When this expansion is no longer possible, a critical cell is created and the expansions then proceed from this new cell. A cell, $\alpha$ is a candidate for homotopic expansion when *num_unpaired_faces*$(\alpha)$ is exactly one; it is a candidate critical cell when all its faces have been paired or inserted in $C$. The priority queues *PQone* and *PQzero*, whose elements are ordered by $G$, store cells with one and zero available faces, respectively. When a cell reaches the front of *PQone* it is paired with its single available face (line 15). If *PQone* is empty, then no candidates for homotopic expansion exist, and the front cell of *PQzero* is added to $C$ as a critical cell.

As illustrative examples, we apply *ProcessLowerStars* to a 2D and a 3D lower star depicted in Figs. 1 and 2, respectively.

The algorithm terminates because there are a finite number of cells, and each cell is inserted in *PQone* at most once. We show in Proposition 4 that Algorithm 1 visits every cell of the lower star and either pairs it or classifies it as critical.

**Proposition 4.** *Each cell in $L(x)$ will be paired and included in $V$ or inserted into $C$ by* ProcessLowerStars.

**Proof.** Every cell in Algorithm 1 that is added to *PQzero* or *PQone* will eventually be either paired as part of a simple homotopy expansion or inserted into $C$. So if $\alpha^{(p)}$ is a cell in $L(x)$ that is omitted from $V$ and $C$, then $\alpha$ will never have been added to *PQzero* or *PQone*. If $\alpha$ is never added to *PQone*, it must be the case that *num_unpaired_faces*$(\alpha)$ is greater that one. This implies that there is a face of $\alpha$, $\delta^{(p-1)} \in L(x)$, that is also omitted from $V$ and $C$. By induction on $p$, there will be a 1-cell in $L(x)$ that is never added to $V$ or $C$. But all 1-cells are added to *PQzero*, giving us a contradiction. □

In the final part of this section, we show that the order cells are inserted into $V$ and $C$ by *ProcessLowerStars* effectively defines a discrete Morse function.

**Proposition 5.** *Let $g$ be a grayscale function defined on a set of voxels, $D$. Let $\mathcal{K}$ be the cubical complex derived from $D$. Assume that $g$ takes distinct values on each voxel of $D$ by adding a tilt if necessary, as in (8). Then, the order in which* ProcessLowerStars *inserts cells into $V$ and $C$ defines a discrete Morse function, as constructed below.*

**Proof.** Let $x$ be a voxel and let $L(x)$ be its lower star. If $L(x)$ has $k > 1$ cells, let $\delta$ be the edge that is minimal with respect to the $G(\alpha)$ ordering (as in Algorithm 1), and if $k > 2$, let $\alpha_1, \alpha_2, \ldots, \alpha_{k-2}$ be the remaining cells in $L(x)$. We record the order that cells from $L(x)$ are inserted into $V$ or $C$ using the convention that if $V(\alpha) = \beta$, then $\beta$ will immediately precede $\alpha$. This algorithm ordering is

$$\delta, x, \alpha_{j_1}, \alpha_{j_2}, \ldots, \alpha_{j_{k-2}} \quad \text{for } j_i \in 1, \ldots, k-2.$$

Let $\epsilon = \min_{x \neq y}\{|g(x) - g(y)|\}$. Note that the maximum number of cells in the lower star of a voxel in a 3D lattice is 27, so we can define a discrete Morse function as follows:

$$m(\delta) = g(x) - \epsilon/30,$$
$$m(x) = g(x),$$
$$m(\alpha_{j_i}) = g(x) + i\epsilon/30.$$

The definition for $m$ extends to all voxels $x \in D$, and all cells $\alpha \in \mathcal{K}$. To prove that $m$ is a discrete Morse function, we verify properties (3) and (4) of Section 2.3. Consider noncritical cells $(\alpha^{(p)}, \beta^{(p+1)}) \in V$: Then, $\alpha, \beta \in L(x)$ for some $x \in D$ and $m(\beta) < m(\alpha)$ from the definition of $m$. All other faces of $\beta$ must have been inserted into $V$ or $C$ at an earlier point in the algorithm, so $\alpha$ is the single face of $\beta$ with greater $m$-value. A critical cell, $\gamma \in C$, has all of its faces inserted earlier in the algorithm and all of its cofaces are added later, so the conditions for a critical cell of a discrete Morse function are also satisfied by $m$. □

### 3.2 Building the Morse Complex

We now describe how to build the Morse chain complex from the discrete vector field defined by $V$ and $C$. The information we require is the incidence relations implied by the boundary operator defined in Section 2.3, (5). This is found simply by following $V$-paths from the faces of each critical cell $\gamma^{(p)} \in C$, and determining which critical $(p-1)$-cells the $V$-paths terminate at.

The algorithm for building the Morse chain complex is shown in Algorithm 2. *ExtractMorseComplex* follows $V$-paths using a breadth-first search algorithm, implemented with a simple queue data structure *Qbfs* (lines 10-20). The breadth-first search is necessary since $V$-paths can branch when followed in the forward direction. $V$-paths can also merge, so, unlike typical search algorithms, we allow multiple traversals of each cell. In addition, this means that the multiplicity of incidence relations can be recorded. Implications for the computational complexity are discussed in Section 3.3.

**Algorithm 2.** ExtractMorseComplex($\mathcal{K}$, $V$, $C$)
**Input:** $\mathcal{K}$, cubical complex.
**Input:** $V$, discrete vector field on $\mathcal{K}$.
**Input:** $C$, critical cells of $V$.
**Output:** $\mathcal{M}$, cells in the Morse chain complex.
**Output:** Facelist, cell adjacencies of the Morse chain complex.

1: **for** $p \in \{0, 1, 2, 3\}$ **do**
2: **for** $\gamma^{(p)} \in C$ **do**
3: create a new $p$-cell $\tilde{\gamma} \in \mathcal{M}$
4: **if** $p > 0$ **then**
5: **for** $\alpha^{(p-1)} < \gamma^{(p)}$ **do**
6: **if** $V[\alpha] \neq \emptyset$ **then**
7: Qbfs.push_back($\alpha$)
8: **end if**
9: **end for**
10: **while** Qbfs $\neq \emptyset$ **do**
11: $\alpha :=$ Qbfs.pop_front
12: $\beta^{(p)} := V[\alpha]$
13: **for** $\delta^{(p-1)} < \beta^{(p)}$ s.t. $\delta \neq \alpha$ **do**
14: **if** $\delta \in C$ **then**
15: add $\tilde{\delta}$ to Facelist($\tilde{\gamma}$)
16: **else if** $V[\delta] \neq \emptyset$ **then**
17: Qbfs.push_back($\delta$)
18: **end if**
19: **end for**
20: **end while**
21: **end if**
22: **end for**
23: **end for**

When adding cell labels to the *Facelist* data structure, we use concatenation, not union or addition modulo 2, to record the multiplicity of incidence relations. Although we will compute homology with $\mathbb{Z}_2$ coefficients, we still require all of the incidence relations between cells to build the chain complex. For example, there may be two distinct $V$-paths from a critical 1-cell to a single critical 0-cell. Taking the union of labels will cause the homology computations to be incorrect, and counting labels modulo 2 will cause us to lose the incidence information entirely. Our approach is consistent with the chain complex defined by (5) and (6).

We define a chain complex from Algorithm 2 as follows: For each $p$-cell $\beta$ in $\mathcal{M}$, define the boundary map $\partial : C_p(X, \mathbb{Z}_2) \rightarrow C_{p-1}(X, \mathbb{Z}_2)$ as

$$\partial\beta = \sum_{\text{critical } \alpha^{(p-1)}} c_{\alpha,\beta}\alpha, \tag{9}$$

where $c_{\alpha,\beta}$ is the number of times that $\alpha$ appears in Facelist($\beta$) (modulo 2). Now the number of times that $\alpha$ appears in Facelist($\beta$) is equal to the number of $V$-paths from $\beta$ to $\alpha$. It follows from Theorem 3 that Algorithm 2 defines a Morse chain complex that calculates the homology of $\mathcal{K}$. Because we can do this at any level of the filtration, we obtain the following result:

**Theorem 6.** *Given a grayscale digital image, $g : D \rightarrow \mathbb{R}$, and its associated cubical complex, $\mathcal{K}$, let $x_i$ be the $i$th voxel in the ordering by grayscale values, $\mathcal{K}_i$ be the lower level cut at the value $g(x_i)$. Suppose that we apply* ProcessLowerStars *and* ExtractMorseComplex *to $\mathcal{K}_i$ and let $\mathcal{M}_i$ be the resulting Morse chain complex. If we use (9) to define the boundary map then the resulting chain complex will calculate the $\mathbb{Z}_2$-homology of $\mathcal{K}_i$.*

The scope of this paper is to analyze 2D and 3D images, and our theorems are proved in this setting. The algorithms *ProcessLowerStars* and *ExtractMorseComplex* can be formulated in more general settings than 3D cubical complexes; for example, functions defined on the vertices of a simplicial complex, or a regular CW complex. Forman's discrete Morse theory applies to any CW-complex [14]. Since the proof of Theorem 6 is a direct application of discrete Morse theory it should be easy to extend *ExtractMorseComplex* and Theorem 6 to more general CW complexes.

### 3.3 Computational Complexity

Algorithm 1 divides the cubical complex $\mathcal{K}$ into $N$ disjoint sets called lower stars, each associated with a single voxel. The lower star $L(x)$ of each voxel $x$ is processed independently; we call this subset of Algorithm 1 *ProcessLowerStar*. Consider a lower star $L(x)$ comprised of $n$ cells. As stated earlier, *ProcessLowerStar* inserts each $\alpha \in L(x)$ into *PQone* exactly once; all cells are popped from *PQone* one at a time.

Therefore, the body of the inner **while** loop is executed exactly $n$ times. All operations are constant time except the priority queue operations, which are logarithmic in the queue size. Since the sizes of *PQone* and *PQzero* are bounded by $n$, then the running time of *ProcessLowerStar* is $O(n \log n)$. Since $n$ for a cubical complex is independent of the image size, $O(n \log n)$ can be considered $O(1)$. Therefore, the complexity of the entire *ProcessLowerStars* algorithm is $O(N)$.

Algorithm 2 traverses all $V$-paths that originate at faces of critical cells, performing simple constant time operations at each cell on these paths. Since the number of critical cells scales with the image size, the body of the **for** loop commencing on line 3 will be executed $O(N)$ times. The **for** loop commencing on line 5 merely checks each face of $\gamma$ and so executes in constant time. The **while** loop commencing on line 10 performs a breadth first search of all of the $V$-paths originating from the faces of $\gamma$. Since each line in the body of the loop is $O(1)$ and since the **for** loop on line 13 runs over the constant number of faces of the cell $\beta$, the body of the **while** loop is $O(1)$. Therefore, Algorithm 2 is $O(M)$, where $M$ is the total number of times the body of the **while** loop is executed. Now, if the $V$-paths were disjoint, the search would visit each cell in $\mathcal{K}$ at most once, so $M < 8N$ and Algorithm 2 would be $O(N)$. However, $V$-paths involving $(\alpha^{(p)}, \beta^{(p+1)})$ pairs (i.e., a $(p, p+1)V$-path) exhibit branching if $p > 0$ and merging if $p < d-1$, where $d$ is the image dimension. To be explicit, in three dimensions, $(0,1)V$-paths can merge, $(1,2)V$-paths can both merge and branch, while $(2,3)V$-paths can only branch.

Branching does not affect of the complexity of Algorithm 2 since it will simply perform a linear-time breadth-first traversal of the branched $V$-path structure (a directed tree) and each cell will only be visited once. On the other hand, $V$-path merging means that a section of $V$-path can be traced backward to more than one originating critical cell. Algorithm 2 visits this section of path multiple times: once for each originating critical cell.

One can construct discrete Morse functions on 2D cubical complexes that contain a single critical 0-cell, but a very large number of (i.e., $O(N)$) critical 1 and 2-cells. If all $V$-paths converge to a single $V$-path some distance from the critical 0-cell, then the multiplicity of this section of path is $O(N)$. If this path section is sufficiently torturous, then its length can also be $O(N)$, so we have $O(N)$ cells being visited $O(N)$ times, yielding a complexity of $O(N^2)$. In 2D, one can recover $O(N)$ complexity by modifying Algorithm 2 to traverse $(0,1)$ $V$-paths in the reverse direction.

In 3D, $(1,2)$ $V$-paths can branch and merge, potentially resulting in many-to-many adjacency relationships between critical 1-cells and critical 2-cells. The pathological 2D example can be modified to produce a 3D Morse function containing $O(N)$ critical 1-cells, each of which connects to $O(N)$ critical 2-cells. This produces the counter-intuitive result that the Morse chain complex of an image with $N$ voxels can contain $O(N^2)$ $V$-paths between critical 1 and 2-cells. It is clearly impossible to construct such a complex in time better than $O(N^2)$. Since the number of critical 1 and 2-cells is bounded by $N$, the number of visits to any cell during the breadth first search is also bounded by $N$ and so the complexity of Algorithm 2 is $O(N^3)$.

We emphasize that the worst cases mentioned above are of mainly theoretical interest and do not affect the utility of this method for large images. While the runtime of Algorithm 2 is $O(N^3)$ for a contrived example, none of the images or objects that we have studied exhibit scaling behavior worse than $O(N)$.

## 4 Proof of Correctness

We want to show that there is a one-to-one correspondence between critical cells identified by *ProcessLowerStars* and changes in the topology of the lower level cuts of the cubical complex. Our proof is broken into two stages. Lemmas 8, 9, and 10 in this section are directly concerned with the behavior of our algorithm on a single lower star; in the Appendix we show how this local analysis is related to the global changes in topology of the lower level cuts. That result (Lemma 12) follows from a standard tool of homology theory (the Mayer-Vietoris sequence) and states that when the closed lower star $\overline{L_i}$ of voxel $x_i$ is added to the filtration $\mathcal{K}_{i-1}$, the changes in homology that occur are determined by the homology of the intersection $\mathcal{K}_{i-1} \cap \overline{L_i}$. This intersection is commonly referred to as the *lower link* or *attachment*.

We now prove that the elements of $H_p(\mathcal{K}_{i-1} \cap \overline{L_i})$ are in one-to-one correspondence with the critical $p$-cells in $L_i$ identified by *ProcessLowerStars*. We start by taking the intersection of the lower star with a suitably small sphere of radius $r$ centered at the voxel $x_i$ and call this subset of $S^2$ the *reduced lower star*, $R_i = L_i \cap S^2(x_i, r)$.

**Lemma 7.** *The reduced lower star, $R_i$, is homotopy equivalent to $\mathcal{K}_{i-1} \cap \overline{L_i}$, and they therefore have the same homology groups.*

**Proof.** The homotopy equivalence is made by growing the sphere used to define $R_i$ until it hits $\mathcal{K}_{i-1} \cap \overline{L_i}$. □

The cell structure of $L_i$ induces a cell structure on $R_i$ so that the map

$$\varphi : L_i \setminus x_i \rightarrow R_i \tag{10}$$

is a bijection of cells. Recall that the full star of a voxel contains eight cubes, 12 square faces, and six edges, so $R_i$ is a subcomplex of the octahedron (which is a simplicial complex of eight triangles, 12 edges, and six vertices). The map $\varphi$ transforms *ProcessLowerStar* into an algorithm that acts on $R_i$ in exactly the same manner; we call this modified algorithm *ProcessReducedStar*. The only difference is the initial pairing of the lowest edge from $L_i$ with the central voxel $x_i$. In *ProcessReducedStar*, the lowest edge of $L_i$ becomes a critical 0-cell in $R_i$ that seeds the growth of a connected component. Cells in $R_i$ are ordered by the same function $G$ derived from the grayscale values of the voxels. Every $(p-1, p)$-pair defined by *ProcessReducedStar* corresponds to a $(p, p+1)$-pair in *ProcessLowerStar*, and every critical $(p-1)$-cell from *ProcessReducedStar* (except for the initial critical 0-cell discussed above) is a critical $p$-cell for *ProcessLowerStar*.

**Algorithm 3.** ProcessReducedStar($R$,$G$)
**Input:** $R$ the reduced lower star.
**Input:** $G$ lexicographic ordering on $R$.
**Output:** $M$ discrete morse function ordering on $R$.

1: $\delta :=$ the 0-cell in $R$ such that $G(\delta)$ is minimal
2: $M := \{\delta\}$ /* $\delta$ *is critical*
3: add all other 0-cells of $R$ to PQzero
4: add all cells $\alpha \in R$ to PQone such that $\alpha > \delta$ and num_unpaired_faces($\alpha$) = 1
5: **while** PQone $\neq \emptyset$ or PQzero $\neq \emptyset$ **do**
6: **while** PQone $\neq \emptyset$ **do**
7: $\alpha :=$ PQone.pop_front
8: **if** num_unpaired_faces($\alpha$) $= 0$ **then**
9: add $\alpha$ to PQzero
10: **else**
11: $M$.append($\alpha, \text{pair}(\alpha)$) /* (*pair*($\alpha$), $\alpha$) *is a pair*
12: remove pair($\alpha$) from PQzero
13: add all cells $\beta \in R$ to PQone such that ($\beta > \alpha$ or $\beta >$ pair($\alpha$)) and num_unpaired_faces($\beta$) $= 1$
14: **end if**
15: **end while**
16: **if** PQzero $\neq \emptyset$ **then**
17: $\gamma :=$ PQzero.pop_front
18: $M$.append($\gamma$) /* $\gamma$ *is critical*
19: add all cells $\alpha \in R$ to PQone such that $\alpha > \gamma$ and num_unpaired_faces($\alpha$) $= 1$
20: **end if**
21: **end while**

We now analyze the algorithm *ProcessReducedStar* and show that the critical $p$-cells it defines are in one-to-one correspondence with elements of the homology group $H_p(R_i)$. We are interested in how cells are added to $M$: either singly when the cell is critical or as a pair $(\alpha, \text{pair}(\alpha))$. Let $B_0 = \{\delta\}$, $B_j =$ {cells in $M$ after $j$ addition steps of the algorithm}. Then $B_j$ is a subcomplex of $R_i$ because $B_0$ is, and at each stage, if we add a pair $(\alpha, \text{pair}(\alpha))$, then all of the faces of $\alpha$ were in $M$ except for $\text{pair}(\alpha)$, which we add now. Alternatively if we add a critical cell, it has come from *PQzero*, so all of its faces were in $M$ by definition. Let $n$ be the total number of addition steps in a call of *ProcessReduced Star* so that $B_n = R_i$.

First note that if $B_{j+1}$ differs from $B_j$ by a pair, then $B_j$ is an elementary collapse of $B_{j+1}$, and so they have the same homotopy type and therefore the same homology and Betti numbers. Thus, the homology can change only when adding a critical cell. Adding a single critical cell of dimension $p$ to the (simplicial) complex $B_j$ can either increase the $p$th Betti number by one (it creates a $p$-cycle) or decrease the $(p-1)$th Betti number by one (it fills in a $(p-1)$-cycle). In the first case, we say the critical cell is positive; in the second, it is negative [30]. Note that a negative critical $p$-cell created in $B_{j+1}$ must pair with and cancel a positive critical $(p-1)$-cell in $B_j$.

The following lemmas show that the number of critical $p$-cells created by *ProcessReducedStar* is exactly the $p$th Betti number of $R_i$. That is, each critical cell identified corresponds to a topological feature in $R_i$, and so each critical cell is positive.

**Lemma 8.** *The number of critical 0-cells is equal to the number of connected components of $R_i$, implying that there are no negative critical 1-cells in the filtration $B_0, \ldots, B_n$.*

**Proof.** We argued above that the homology of $B_{j+1}$ can differ from that of $B_j$ only when they differ by a critical cell. Therefore, the number of critical 0-cells is at least the number of connected components of $R_i$.

Suppose now that a single connected component $A \subset R_i$ has two critical 0-cells, $p$ and $q$, with $G(p) < G(q)$. Let $j$ be the index of the step in which the critical cell $q$ is added. This means $p \in B_{j-1}$ and there are no cells adjacent to $B_{j-1}$ with a single unpaired face, i.e., *PQone* is empty. Now since $p, q \in A$ and $A$ is connected, there is a path of vertices and edges in $A$, $p < e_1 > x_1 < e_2 > x_2 < \ldots e_k > x_k = q$. But the existence of this path implies there is an edge $e_l$ with one vertex, $x_{l-1} \in B_{j-1} \cap A$ and the other vertex $x_l \notin B_{j-1}$, implying that $e_l$ should be in *PQone*, so we have a contradiction. Therefore, there is exactly one critical 0-cell for each connected component of $R_i$. □

**Lemma 9.** *Each critical 1-cell corresponds to a nonbounding cycle in $R_i$, implying that there are no negative critical 2-cells in the filtration $B_0, \ldots, B_n$.*

**Proof.** First assume that $R_i$ has a nonbounding 1-cycle. In the filtration $B_0, \ldots, B_n$, where $B_n = R_i$, there must be a step $B_j$ to $B_{j+1}$ where $B_j$ has the same homotopy as $B_0$—a single point—and $B_{j+1}$ has a nonbounding 1-cycle. The only possible way to achieve this is through the addition of a critical 1-cell.

Conversely, suppose that the algorithm has reached step $B_j$ and that the next step is to add the critical 1-cell, $\gamma$. Both vertices of $\gamma$ are in $B_j$; in fact, both vertices of $\gamma$ are in the same connected component, $B'$ of $B_j$, since otherwise there would be a connected component of $R_i$ with more than one critical 0-cell. Therefore, there must be a 1-cycle that passes through $\gamma$. Choose $z_\gamma$ to be a shortest possible 1-cycle passing through $\gamma$. By construction, $z_\gamma$ is nonbounding for $B' \cup \gamma$ because neither cofaces of $\gamma$ are in $B'$. Let $A$ be the connected component of the lower link that contains $B' \cup \gamma$. We want to show that $z_\gamma$ is nonbounding for $A$, and we will do this by assuming otherwise and obtaining a contradiction. So assume $z_\gamma$ is the boundary of a 2-chain, $w \subset A$. Adjust $z_\gamma$ and $w$ if necessary so that $z'_\gamma$ is homologous to $z_\gamma$ in $B' \cup \{\gamma\}$ and $w' \subset A \setminus B'$. Note that exactly one coface $\alpha > \gamma$ is in $w'$. If $w' = \alpha$, we immediately have a contradiction since $\alpha$ then has a single free (unpaired) face, $\gamma$, with respect to $B'$. So $w'$ has at least two 2-cells, $\alpha, \beta$, and they may be chosen to share a face in $A \setminus \{B' \cup \gamma\}$. Now consider the graph $\Gamma$ whose vertices are the 2-cells in $w'$ and edges match those 1-cells $\delta \in A \setminus \{B' \cup \gamma\}$ such that $\delta$ is the common face of a pair of 2-cells in $w'$. $\Gamma$ is connected because $z'_\gamma$ is a shortest possible cycle through $\gamma$. If there is a loop in $\Gamma$, then the corresponding 2-cells surround a 0-cell in $A \setminus \{B' \cup \gamma\}$, but this 0-cell must be connected to $B'$ by some path, which implies the existence of a free pair for $B'$ (as in the proof of Lemma 8). Therefore, $\Gamma$ must be a tree, and since $\Gamma$ has at least two vertices, it must have an edge with a dangling vertex $\sigma \neq \alpha$. Then, $\sigma$ is a 2-cell with all of its faces except one in $B'$, and this contradicts our assumption that there were no free pairs for $B'$. □

**Lemma 10.** *A critical 2-cell occurs only when $R_i$ is the entire octahedron.*

**Proof.** Suppose we have reached the complex $B_j$ in the filtration, and that the next step is the addition of a critical 2-cell, $\gamma$. By Lemma 2, $\gamma$ cannot be a negative critical 2-cell, so $B_j \cup \gamma$ must contain a 2-cycle. But the only possible 2-cycle in $S^2$ is the whole of $S^2$.

Now suppose $R_i$ is the full octahedron. Let $\gamma$ be any 2-cell in $R_i$ and consider the subcomplex $A = R_i \setminus \gamma$. The action of our algorithm on $A$ is such that it will create a single critical 0-cell for the initial vertex and no other critical cells. Then, $\gamma$ is a cell with all its faces in $A$ and so it will become a critical 2-cell. □

Lemmas 8, 9, and 10 show that the number of critical $p$-cells identified by *ProcessReducedStar* is exactly the rank of $H_p(R_i)$, and Lemma 7 tells us that this is the same as the rank of $H_p(\mathcal{K}_{i-1} \cap \overline{L_i})$. The result in the Appendix, Lemma 12, shows how elements of $H_p(\mathcal{K}_{i-1} \cap \overline{L_i})$ correspond to changes in the homology of successive lower level cuts, $\mathcal{K}_{i-1}$ and $\mathcal{K}_i$. We bring all of these results together in the following theorem.

**Theorem 11.** *Given a grayscale digital image, $g : D \to \mathbb{R}$, and its associated cubical complex, $\mathcal{K}$, let $x_i$ be the $i$th voxel in the ordering by grayscale values, $\mathcal{K}_i$ be the lower level cut at the value $g(x_i)$, and $L_i$ be the lower star of $x_i$. Then, for each critical $p$-cell $\gamma^{(p)} \in L_i$ identified by* ProcessLowerStars *either*

1. *$\gamma$ creates a new $p$-cycle in $H_p(\mathcal{K}_i)$ or*
2. *$\gamma$ fills in a $(p-1)$-cycle from $H_{p-1}(\mathcal{K}_{i-1})$.*

We finish this section with a discussion of the wider applicability of our results. The proofs of Lemmas 8, 9, and 10 do not make specific use of the octahedral structure of the lower link, so our Theorem 11 immediately generalizes to any 3D regular cell complex with a function defined on its vertices. Regular cell complexes include simplicial complexes, and CW complexes where the attaching maps are homeomorphisms (not just continuous maps). It is also easy to adapt Lemmas 8, 9, and 10 to 1D and 2D regular cell complexes. Higher-dimensional cell complexes present a greater challenge. Couprie and Bertrand [3], [4] prove some related results about 4D cubical complexes and we think it is likely that Theorem 11 will hold in four dimensions also. Proving this, however, will require different techniques to those we have used in this paper. Couprie and Bertrand [3], [4] also show that their results fail to hold in five dimensions and higher due to potential obstructions in finding free pairs for the simple homotopy collapse. Our *ProcessLowerStars* algorithm is likely to create extra critical cells in these situations and Theorem 11 may no longer hold in five dimensions and higher.

## 5 Applications

We have successfully applied the algorithms from Section 3 to both 2D and 3D grayscale images. In this section, we present two examples—a 2D section of sandstone and a 3D image of a metal foam—for which the persistent Betti numbers are computed. Both of these images were obtained using the X-ray microcomputed tomography facility at the Australian National University [31].

The first step in each case is to construct the discrete Morse function from the grayscale image and then build the Morse chain complex, $\mathcal{M}$. We also determine a filtration of the Morse complex by assigning the grayscale value of the critical cell to its corresponding cell in $\mathcal{M}$. If two critical cells have the same grayscale value, we order them by their dimension or in arbitrary order if they have the same dimension. We then have a total ordering of cells in $\mathcal{M}$, as required for persistent Betti number computations. We use Zomorodian's *Pair-Cells* algorithm from [32, page 22] to compute the persistent Betti numbers.

Our implementation of these algorithms is in *Python* and the code ran on a Dell Precision 390 PC (2400 MHz Core 2 Duo CPU with 2 GB RAM). The implementation in *Python* is a prototype; we expect the algorithm would be faster if optimized and implemented in a different language. The timings we present are therefore for comparison only.

### 5.1 Example: 2D Image

We apply the algorithms *ProcessLowerStars* and *Extract MorseComplex* to the 2D image of a section of rock shown in Fig. 3a. The dimensions of the data are $200 \times 200$ pixels, so the cubical complex has 159,201 cells ($200 \times 200$ 0-cells, $200 \times 199 \times 2$ 1-cells, $199 \times 199$ 2-cells). The range of the image is limited to 256 different possible grayscale values, so we apply the tie-breaking scheme given in (8) before ordering the voxels (black pixels are low, white pixels are high). The Morse complex, Fig. 3b, has 2,827 cells in total (1.8 percent of the cubical complex)—a dramatic simplification of the combinatorial structure. It took 16 seconds for the algorithm to generate the Morse complex.

We obtain topological information about the image by computing the persistent Betti numbers with persistence measured as the difference in grayscale function values of the creator and destroyer critical cells. Plots of persistent Betti numbers are shown in Fig. 4. We were able to verify that the original cubical complex had the same Betti numbers. It took 6 seconds to calculate the persistent Betti numbers of the Morse complex, and 882 seconds to calculate the persistent Betti numbers of the cubical complex.

We can see from Fig. 4 that there are three 0-cycles with persistence greater than or equal to 96 and just one with persistence greater than or equal to 112. Because we pair cells, it is possible to identify these cycles. The most persistent of these cycles (apart from the 0-cycle of infinite persistence) is illustrated in Fig. 3c.

Fig. 4 also shows that there are two 1-cycles with persistence greater or equal to 64, and none with persistence greater than or equal to 80. One of the highly persistent 1-cycles is illustrated in Fig. 3d.

### 5.2 Example: 3D Foam

We now study a 3D image of a metal foam; a visualization of the data is shown in Fig. 5. The dimensions of the data are $64 \times 64 \times 64$ voxels, and the cubical complex has 1,540,351 cells. The Morse complex has 4,093 cells (0.27 percent of the cubical complex); it took 425 seconds for our algorithms to generate this complex.

The persistent Betti numbers describe the topology of the porespace of the foam; plots of the $\lambda$-persistent Betti numbers are displayed in Fig. 6, with persistence again measured as the difference in grayscale values of the creator and destroyer critical cells. For intensities less than 1,800 we see a sharp spike in all of the Betti numbers; this is due to random, small-scale modulations in the porespace of the

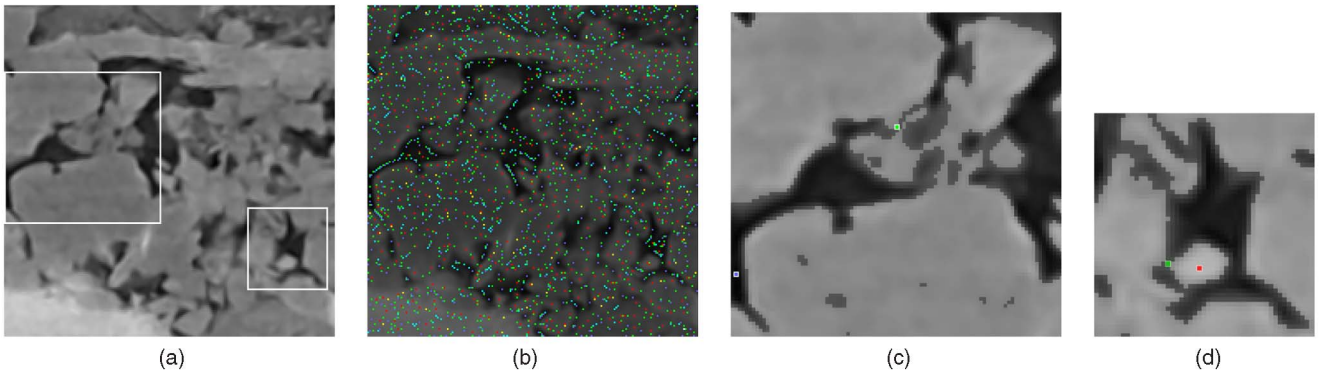


Fig. 3. (a) A $200 \times 200$ pixel image of a section of sandstone. The white rectangles mark the regions shown in (c) and (d). (b) Locations of the critical cells of the discrete Morse function derived from (a) by applying *ProcessLowerStars*. Pixels are blue if they represent critical 0-cells, green for critical 1-cells, and red for critical 2-cells. If a pixel has more than one type of critical cell at its coordinate, then the colors are added. (c) A subset from the upper left of (a). Two critical cells are marked: a blue pixel (lower left) that corresponds to the critical 0-cell that creates a 0-cycle and a green pixel (center) that marks the critical 1-cell that destroys this 0-cycle. The dark pixels belong to the lower level cut at the value of the critical 1-cell. When the green pixel is added to the dark pixels, the connected component corresponding to the 0-cycle joins with the connected component on the right of the image, "destroying" the 0-cycle on the left. (d) A subset from the lower right of (a). Two critical cells are marked: a green pixel (left) that creates a 1-cycle and a red pixel (center) that marks the critical 2-cell which destroys the 1-cycle. The dark pixels belong to the lower level cut at the value of the critical 1-cell. When the green pixel is added to the dark pixels, a hole is created in the lower level cut. This hole is filled in when the level cut threshold reaches the value of the red pixel.

foam, as can be seen from the transparent to pink transition in Fig. 5. When the intensity threshold is around 2,000, we see that Betti-0 equals one, signaling that the porespace is connected. At the same threshold, Betti-1 equals 24 and Betti-2 is zero, so the porespace is an "open-celled foam." The highly persitent Betti-1 values between intensity thresholds of 2,000 and 5,000 are due to closed paths in the porespace that in turn signal the presence of "handles" in the material that the foam is made from.

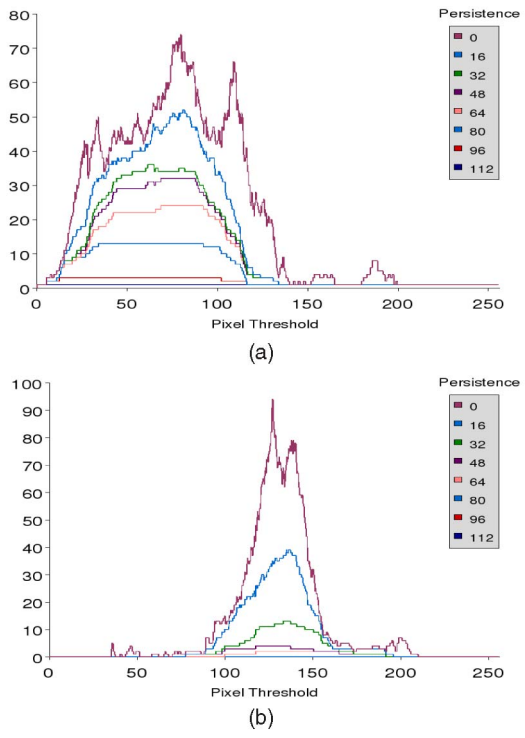


Fig. 4. The $\lambda$-Persistent Betti 0 (a) and Betti 1 (b) numbers of lower level cuts of the sandstone image as a function of threshold and for a sequence of $\lambda$-values.

## APPENDIX

Here, we prove that changes in the homology of successive lower level cuts are determined by the structure of the intersection of the lower star with the existing complex. The result applies more generally than this specific context, so we state the minimal conditions necessary in the lemma, but preserve the notation of Section 4.

**Notation.** *Recall that an element of a homology group is an equivalence class of p-cycles, and is denoted* $[z_p] \in H_p(\mathcal{K})$, *where* $z_p$ *is some representative of the class.*

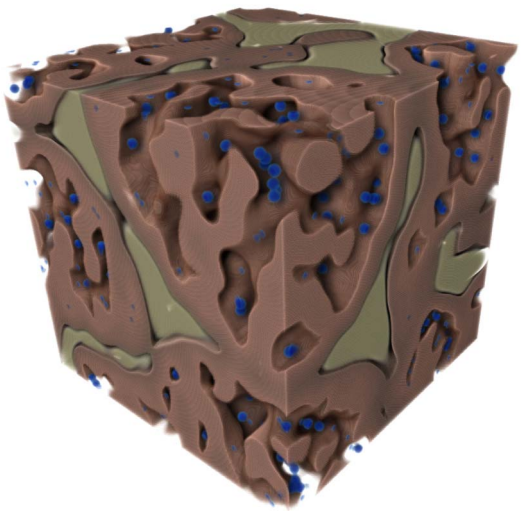

Fig. 5. Visualization (using *Drishti* [33]) of a 3D foam and the location of minima (blue dots) as identified using *ProcessLowerStars*. Some level cuts of the grayscale image are rendered as follows: The darkest voxels (intensity values 0-1,000) are transparent and show the noisy low-persistence structures discussed in the text; slightly lighter voxels (1,000-3,500) are colored pinkish brown; the brightest voxels are yellow. A rotating view of this data may be viewed in the online supplementary material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2011.95.

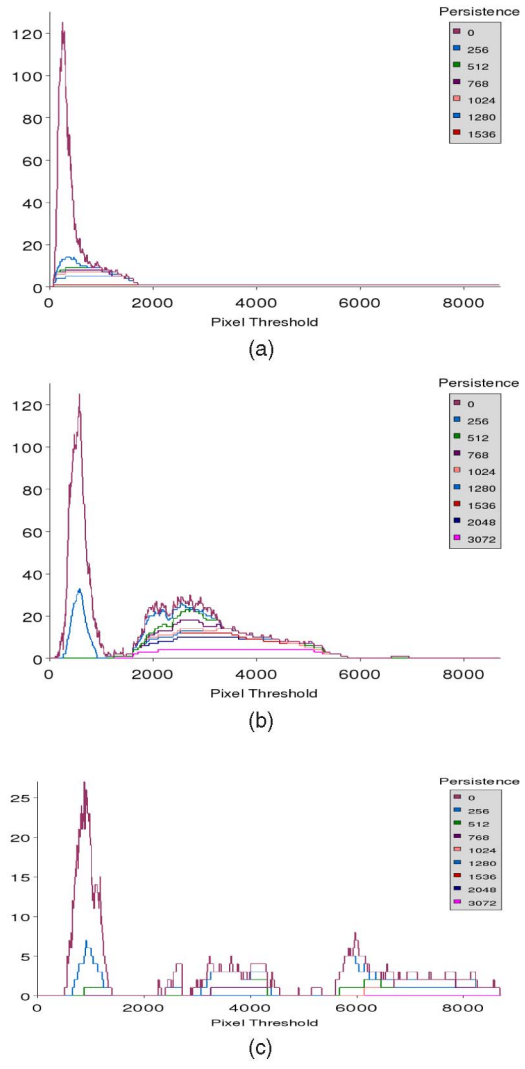


Fig. 6. The $\lambda$-persistent Betti 0 (a) Betti 1 (b) and Betti 2 (c) numbers for the 3D foam for a sequence of $\lambda$ values.

**Lemma 12.** *Given three spaces, $\mathcal{K}_i$, $\mathcal{K}_{i-1}$, and $\overline{L_i}$ such that $\mathcal{K}_i = \mathcal{K}_{i-1} \cup \overline{L_i}$, where $\overline{L_i}$ is a closed contractible space, consider the map of homology groups $\pi_p : H_p(\mathcal{K}_{i-1}) \to H_p(\mathcal{K}_i)$ induced by the inclusion of $\mathcal{K}_{i-1}$ into $\mathcal{K}_i$. Then, for each element of $H_{p-1}(\mathcal{K}_{i-1} \cap \overline{L_i})$, we find either*

1. *a $p$-cycle $[z_p] \in H_p(\mathcal{K}_i)$ with $[z_p] \notin \mathrm{Im}\pi_p$ or*
2. *a $(p-1)$-cycle $[z_{p-1}] \in \mathrm{Ker}\pi_{p-1}$.*

That is, any change in homology between $\mathcal{K}_{i-1}$, and $\mathcal{K}_i$ is signaled by the presence of a nonbounding cycle in $\mathcal{K}_{i-1} \cap \overline{L_i}$.

**Proof.** We begin with the Mayer-Vietoris sequence of homology groups:

$$\cdots \xrightarrow{j_p} H_p(\mathcal{K}_{i-1}) \oplus H_p(\overline{L_i}) \xrightarrow{s_p} H_p(\mathcal{K}_i) \xrightarrow{v_p} H_{p-1}(\mathcal{K}_{i-1} \cap \overline{L_i}) \xrightarrow{j_{p-1}} H_{p-1}(\mathcal{K}_{i-1}) \oplus H_{p-1}(\overline{L_i}) \xrightarrow{s_{p-1}} \cdots \quad (11)$$

The homomorphisms are defined as follows:

$$j_p([u]) = ([u], -[u]),$$
$$s_p([w], [w']) = [w + w'],$$
$$v_p([z]) = [\partial z'],$$

where, in the last equation, $z$ is a cycle in $\mathcal{K}_i$ and we can write $z = z' + z''$, where $z'$ and $z''$ are chains (not necessarily cycles) in $\mathcal{K}_{i-1}$ and $\overline{L_i}$, respectively. These homomorphisms are well defined (see, for example, [26, Theorem 33.1]). The crucial property of the Mayer-Vietoris sequence is that it is *exact*—the image of each homomorphism is equal to the kernel of the following one: $\mathrm{Im}\ j_p = \mathrm{Ker}\ s_p$, $\mathrm{Im}\ s_p = \mathrm{Ker}\ v_p$, and $\mathrm{Im}\ v_p = \mathrm{Ker}\ j_{p-1}$.

Now, $\overline{L_i}$ is contractible, so its homology groups are the same as those of a single point and the above sequence becomes

$$\cdots \xrightarrow{j_p} H_p(\mathcal{K}_{i-1}) \xrightarrow{\pi_p} H_p(\mathcal{K}_i) \xrightarrow{v_p} H_{p-1}(\mathcal{K}_{i-1} \cap \overline{L_i}) \xrightarrow{j_{p-1}} H_{p-1}(\mathcal{K}_{i-1}) \cdots \quad (12)$$

for $p \geq 1$ and the following for $p = 0$:

$$\cdots H_0(\mathcal{K}_{i-1} \cap \overline{L_i}) \xrightarrow{j_0} H_0(\mathcal{K}_{i-1}) \oplus \mathbb{Z} \xrightarrow{s_0} H_0(\mathcal{K}_i) \longrightarrow 0. \quad (13)$$

Notice that $j_p$ and $s_p$ are now simply inclusion-induced homomorphisms (except when $p = 0$) so $s_p = \pi_p$ and we have that $\mathrm{Im}\ \pi_p = \mathrm{Ker}\ v_p$ and $\mathrm{Ker}\ \pi_p = \mathrm{Im}\ j_p$. This tells us that the changes in homology that occur when $\overline{L_i}$ is added are determined by the homology of the intersection $\mathcal{K}_{i-1} \cap \overline{L_i}$. The remainder of our proof proceeds by showing that each element of $H_{p-1}(\mathcal{K}_{i-1} \cap \overline{L_i})$ corresponds either to the creation of a new $p$-cycle in $H_p(\mathcal{K}_i)$ or the filling in of a $(p-1)$-cycle of $H_{p-1}(\mathcal{K}_{i-1})$.

For $p \geq 1$, consider a nontrivial $(p-1)$-cycle $[z] \in H_{p-1}(\mathcal{K}_{i-1} \cap \overline{L_i})$; $z$ is either the boundary of a $p$-chain in $\mathcal{K}_{i-1}$ or it is not. In the case that it is a boundary, $z \in \mathrm{Ker}\ j_{p-1}$ and so $z \in \mathrm{Im}\ v_p$ by exactness. Therefore, there is some $[w] \in H_p(\mathcal{K}_i)$ with $v_p([w]) = [z] \neq 0$, so $[w] \notin \mathrm{Ker}\ v_p$. Exactness again implies that $[w] \notin \mathrm{Im}\pi_p$. This means that there is no cycle from $H_p(\mathcal{K}_{i-1})$ that maps onto $[w]$—or, in other words, $[w]$ is a new homology class in $H_p(\mathcal{K}_i)$.

On the other hand, for $p > 1$, suppose that $[z] \in H_{p-1}(\mathcal{K}_{i-1} \cap \overline{L_i})$ is not the boundary of any $p$-chain in $\mathcal{K}_{i-1}$. Then, $j_{p-1}([z])$ is a nontrival homology class of $H_{p-1}(\mathcal{K}_{i-1})$, but since $\mathrm{Im}j_{p-1} = \mathrm{Ker}\ \pi_{p-1}$, we must have $[z]$ homologous to zero in $H_{p-1}(\mathcal{K}_i)$. Thus, adding $\overline{L_i}$ destroys a $(p-1)$-cycle.

Finally, if $p = 1$ and $[z] \in H_{p-1}(\mathcal{K}_{i-1} \cap \overline{L_i})$ is not the boundary of any 1-chain in $\mathcal{K}_{i-1}$, we must consider the end of the Mayer-Vietoris sequence given in (13). There are three possibilities to consider:

1. $H_0(\mathcal{K}_{i-1} \cap \overline{L_i})$ has $k > 1$ connected components. Then, let $z = z_1 - z_2$ be a zero cycle that is not the boundary of any 1-chain from $\mathcal{K}_{i-1}$. Since $\overline{L_i}$ is connected, $z \sim 0 \in H_0(\overline{L_i})$ and therefore $z \sim 0 \in H_0(\mathcal{K}_i)$, and we have destroyed a 0-cycle.
2. $H_0(\mathcal{K}_{i-1} \cap \overline{L_i}) = [z]$ for a single vertex $z \in \mathcal{K}_{i-1} \cap \overline{L_i}$. Then adding $\overline{L_i}$ does not change the number of connected components so $H_0(\mathcal{K}_i) = H_0(\mathcal{K}_{i-1})$.

3. $\mathcal{K}_{i-1} \cap \overline{L_i} = \emptyset$, so $\overline{L_i}$ creates a new connected component, i.e., a new homology class for $H_0(\mathcal{K}_i)$. □

## ACKNOWLEDGMENTS

This work was supported by the Australian Research Council Discovery project DP0666442. We would like to thank Ajay Limaye and Munish Kumar for their help rendering Fig. 5 using the visualization tool *Drishti*.

## REFERENCES

[1] V.A. Kovalevsky, "Finite Topology as Applied to Image Analysis," *Computer Vision, Graphics, and Image Processing,* vol. 46, no. 2, pp. 141-161, 1989.
[2] M. Couprie, F.N. Bezerra, and G. Bertrand, "Topological Operators for Grayscale Image Processing," *J. Electronic Imaging,* vol. 10, no. 4, pp. 1003-1015, 2001.
[3] M. Couprie and G. Bertrand, "New Characterizations of Simple Points, Minimal Non-Simple Sets and P-Simple Points in 2D, 3D and 4D Discrete Spaces," *Proc. 14th IAPR Int'l Conf. Discrete Geometry for Computer Imagery,* pp. 105-116, 2008.
[4] M. Couprie and G. Bertrand, "New Characterizations of Simple Points in 2D, 3D and 4D Discrete Spaces," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 31, no. 4, pp. 637-648, Apr. 2009.
[5] H. Edelsbrunner, D. Letscher, and A. Zomorodian, "Topological Persistence and Simplification," *Discrete Computational Geometry,* vol. 28, pp. 511-533, 2002.
[6] D. Laney, P.-T. Bremer, A. Mascarenhas, P. Miller, and V. Pascucci, "Understanding the Structure of the Turbulent Mixing Layer in Hydrodynamic Instabilities," *IEEE Trans. Visualization and Computer Graphics,* vol. 12, no. 5, pp. 1053-1060, Sept./Oct. 2006.
[7] L. Vincent and P. Soille, "Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 13, no. 6, pp. 583-598, June 1991.
[8] R. Jones, "Connected Filtering and Segmentation Using Component Trees," *Computer Vision and Image Understanding,* vol. 75, pp. 215-228, 1999.
[9] A. Cayley, "On Contour and Slope Line," *The Philosophical Magazine,* vol. 18, no. 120, pp. 264-268, 1859.
[10] J.C. Maxwell, "On Hills and Dales," *The Philosophical Magazine,* vol. 40, no. 269, pp. 421-427, 1870.
[11] J. Milnor, *Morse Theory, (Annals of Mathematical Studies 51).* Princeton Univ. Press, 1969.
[12] H. Edelsbrunner, J. Harer, and A. Zomorodian, "Hierarchical Morse Complexes for Piecewise Linear 2-Manifolds," *Proc. 17th Ann. Symp. Computational Geometry,* pp. 70-79, 2001.
[13] S. Biasotti, L. De Floriani, B. Falcidieno, P. Frosini, D. Giorgi, C. Landi, L. Papaleo, and M. Spagnuolo, "Describing Shapes by Geometrical-Topological Properties of Real Functions," *ACM Computing Surveys,* vol. 40, no. 4, pp. 1-87, 2008.
[14] R. Forman, "Morse Theory for Cell Complexes," *Advances in Math.,* vol. 134, pp. 90-145, 1998.
[15] R. Forman, "A User's Guide to Discrete Morse Theory," *Séminaire Lotharingien de Combinatoire,* vol. 48, 2002.
[16] T.F. Banchoff, "Critical Points and Curvature for Embedded Polyhedral Surfaces," *The Am. Math. Monthly,* vol. 77, no. 5, pp. 475-485, http://www.jstor.org/stable/2317380, 1970.
[17] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci, "Morse-Smale Complexes for Piecewise Linear 3-Manifolds," *Proc. 19th Ann. Symp. Computational Geometry,* pp. 361-370, 2003.
[18] A. Gyulassy, V. Natarajan, V. Pascucci, and B. Hamann, "Efficient Computation of Morse-Smale Complexes for Three-Dimensional Scalar Functions," *IEEE Trans. Visualization and Computer Graphics,* vol. 13, no. 6, pp. 1440-1447, Nov./Dec. 2007.
[19] T. Lewiner, H. Lopes, and T. Geovan, "Optimal Discrete Morse Functions for 2-Manifolds," *Computational Geometry: Theory and Applications,* vol. 26, pp. 221-233, 2003.
[20] T. Lewiner, H. Lopes, and T. Geovan, "Towards Optimality in Discrete Morse Theory," *Experimental Math.,* vol. 12, pp. 271-285, 2003.
[21] T. Lewiner, "Geometric Discrete Morse Complexes," PhD dissertation, Pontifícia Universidade Católica do Rio de Janeiro, 2005.
[22] H. King, K. Knudson, and N. Mramor, "Genering Discrete Morse Functions from Point Data," *Experimental Math.,* vol. 14, pp. 435-444, 2005.
[23] A. Gyulassy, P.-T. Bremer, B. Hamann, and V. Pascucci, "A Practical Approach to Morse-Smale Complex Computation: Scalability and Generality," *IEEE Trans. Visualization and Computer Graphics,* vol. 14, no. 6, pp. 1619-1626, Nov./Dec. 2008.
[24] A. Gyulassy, "Combinatorial Construction of Morse-Smale Complexes for Data Analysis and Visualization," PhD dissertation, Univ. of California, Davis, Dec. 2008.
[25] A. Hatcher, *Algebraic Topology.* Cambridge Univ. Press, Nov. 2001.
[26] J.R. Munkres, *Elements of Algebraic Topology.* Perseus Books Publishing, 1984.
[27] V. Robins, "Towards Computing Homology from Finite Approximations," *Proc. Topology* , vol. 24, pp. 503-532, 1999.
[28] J.H.C. Whitehead, "Simplicial Spaces, Nuclei, and $m$-Groups," *Proc. London Math. Soc.,* vol. 45, no. 2, pp. 243-327, 1939.
[29] R. Bott, *Morse Theory Indomitable,* vol. 68, pp. 99-114. Publications Mathématiques de l'I.H.É.S., 1988.
[30] C.J.A. Delfinado and H. Edelsbrunner, "An Incremental Algorithm for Betti Numbers of Simplicial Complexes on the 3-Sphere," *Computer Aided Geometric Design,* vol. 12, pp. 771-784, 1995.
[31] A. Sakellariou, T.J. Senden, T.J. Sawkins, M.A. Knackstedt, A. Limaye, C.H. Arns, A.P. Sheppard, and R.M. Sok, "An X-Ray Tomography Facility for a Wide Range of Mesoscale Physics Applications," *Proc. SPIE,* vol. 5535, pp. 166-171, 2004.
[32] A. Zomorodian, "Computational Topology," *Algorithms and Theory of Computation Handbook,* second ed., M. Atallah and M. Blanton, eds., Chapman & Hall/CRC Press, 2009.
[33] *Drishti—Volume Exploration and Presentation Tool,* Ser. Visualization. Baltimore: IEEE, Poster Presentation, 2006.

**Vanessa Robins** received the BSc (Hons) degree in mathematics from the Australian National University in 1994 and received the PhD degree in applied mathematics from the University of Colorado, Boulder, in 2000. She has been a research fellow in the Department of Applied Mathematics at the Australian National University since 2000. Her research interests are in computational topology and geometry, and the enumeration and classification of periodic nets. She works on mathematical foundations, algorithms, and scientific applications in both of these fields.

**Peter John Wood** received the BSc (Hons) degree from the Australian National University in 1996 and the PhD degree from the Flinders University of South Australia in 2003. Between 2005 and 2006, he worked as a programmer for Hunting Tank Software. Since 2007, he has been a postdoctoral fellow at the Australian National University. His research interests include computational topology, operator algebras, wavelets, and some mathematical problems associated with climate change mitigation.

**Adrian P. Sheppard** received the BSc (Hons) degree in mathematical physics from The University of Adelaide in 1992 and the PhD degree in physics from the Australian National University in 1996. He has held research positions at the Université Libre de Bruxelles, the University of New South Wales, and the Australian National University, where he has worked since 2000. His research interests include the characterization of shape, fluid flow in porous media, parallel algorithms for image processing, and tomographic imaging.