

TIGHTENING: MORPHOLOGICAL SIMPLIFICATION

JASON WILLIAMS* and JAREK ROSSIGNAC†

*College of Computing, Georgia Institute of Technology
85 5th St. NW, Atlanta, Georgia 30332, USA*

**jasonw@cc.gatech.edu*

†jarek@cc.gatech.edu

Received 1 November 2004

Revised 15 August 2005

Communicated by Leif Kobbelt and Vadim Shapiro, Guest Editors

ABSTRACT

Given a two- or three-dimensional set S of arbitrary topology and a radius r , we show how to construct an r -tightening of S , which is a set whose boundary has mean curvature with magnitude less than or equal to $1/r$ and which only differs from S in a morphologically-defined tolerance zone we call the mortar. The mortar consists of the thin or highly curved parts of S and its complement, such as corners, gaps, and small connected components, while the boundary of a tightening consists of components of locally minimal length (in 2D) or area (in 3D) that lie in the mortar. Tightenings are defined independently of shape representation, and it may be possible to find them using a variety of algorithms. We describe how to approximately compute tightenings for two-dimensional sets represented as binary images and for three-dimensional sets represented as triangle meshes using constrained, level-set curvature flow.

Keywords: Mathematical morphology; topology; simplification.

1. Introduction

We focus our discussion and implementation on full-dimensional subsets of \mathbb{R}^2 or \mathbb{R}^3 . Our input is a set S that is the closure of its interior. Let ∂S denote the boundary of S , iS its interior, and $\neg S$ its complement.

The complexity of S may be measured in a variety of ways. The tightening process described in this paper decreases several of these measures. In 2D, it increases smoothness (the minimum radius of curvature of ∂S), while in 3D it limits the mean curvature. Furthermore, it tends to reduce topological complexity (the genus and number of components of S and $\neg S$), increase the least feature size (the minimum distance between ∂S and its medial axis,) and decrease perimeter or surface area. Tightening does not modify the r -regular portions of ∂S , which lie further than r from the medial axis of ∂S . Instead, it confines its effect to the mortar, which is typically a small subset of the more commonly used tolerance zone $Z_r(S)$ defined as the set of all the points within a distance r of ∂S .

We begin by providing a precise definition of r -tightenings and then describe how to compute approximations of r -tightenings for input sets represented as binary images in 2D or triangle meshes in 3D. We conclude with a discussion of tightenings in relation to prior art.

2. Definitions

2.1. r -Smoothness

In 2D, a G^1 -continuous curve is r -smooth if and only if the absolute value of its radius of curvature is greater than or equal to r at all points of the curve where it is defined. The intent of this definition is to allow curves with discontinuous curvature to be considered r -smooth, provided they are tangent-continuous and the absolute value of the radius of curvature to either side of a curvature discontinuity is greater than or equal to r .

In 3D, a surface is r -smooth if the absolute values of its principal radii of curvature are greater than or equal to r where they are defined. A full-dimensional set (planar region in 2D or solid in 3D) is r -smooth if and only if its boundary is r -smooth.

2.2. Signed curvature

We adopt the convention that a curvature value at a point on the boundary of a full-dimensional set is negative if the associated osculating circle lies to the outward side of the set and positive if it lies on the inward side.

2.3. Growing and shrinking

The mortar is defined in terms of the morphological operations of rounding and filleting, which can in turn be defined in terms of growing and shrinking.

S grown by r , denoted $S \uparrow_r$, is the set of all points whose distance from S is less than r (Fig. 1 (a)).¹ It does not include its boundary, so it is topologically open. S shrunk by r , denoted $S \downarrow_r$, is the set of all points whose distance from $\neg S$ is greater than or equal to r (Fig. 1 (b)). It includes its boundary, so it is closed. Shrinking eliminates small or thin pieces of S , while growing fills in gaps and cracks. Both operations can change the topology of S by, for example, changing the number of components of S or $\neg S$.

We can highlight properties of growing and shrinking by providing alternative definitions equivalent to those given above. If we let B_r denote an open disk of radius r , then $S \uparrow_r$ is the union of all B_r whose centers lie in S , while $S \downarrow_r$ is the complement of the union of all B_r whose centers lie in $\neg S$. This formulation shows that growing and shrinking are dual: $\neg(S \uparrow_r) = (\neg S) \downarrow_r$, so that, for instance, the same result may be achieved by growing a set as by shrinking its complement. It also implies that, in two dimensions, the signed curvature of $\partial(S \uparrow_r)$ is everywhere

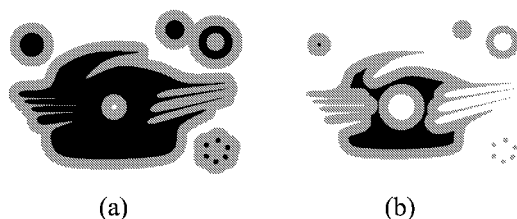


Fig. 1. (a) The input set S is shown in black, and the material $S \uparrow_r$ adds to S is shown in gray. (b) $S \downarrow_r$ is shown in black, and the material it removes from S is shown in gray.

less than or equal to $1/r$, while symmetrically, the curvature of $\partial i(S \downarrow_r)$ is everywhere greater than or equal to $-1/r$. $S \downarrow_r$ can have lower-dimensional portions, but because these can be approached by a disk of radius r on both sides, we can say the magnitude of their curvature is less than $1/r$. In three dimensions the maximum principal curvature of $\partial(S \uparrow_r)$ is less than $1/r$, while the minimum curvature of $\partial i(S \downarrow_r)$ is greater than $-1/r$. We can also define growing and shrinking using the tolerance zone $Z_r(S)$ around ∂S defined as the union of all B_r whose centers lie on ∂S . Equivalently, $Z_r(S) = (\partial S) \uparrow_r$. Then $S \downarrow_r$ is $S - Z_r(S)$, while $S \uparrow_r$ is $S \cup Z_r(S)$.

2.4. Rounding and filleting

S rounded by r , denoted $R_r(S)$, is $S \downarrow_r \uparrow_r$ (Fig. 2 (a)). $R_r(S)$ is the union of all B_r that lie in S . This can be seen by considering that $S \downarrow_r$ can also be described as the centers of all B_r that lie in S . When we grow the centers of the disks, we obtain the disks themselves. It follows that, like $S \uparrow_r$, $R_r(S)$ is open and has maximum curvature less than or equal to $1/r$. We can also conclude that $R_r(S) \subseteq S$. The set $S - R_r(S)$ contains the points of S that cannot be included in any B_r in S disjoint from ∂S . It includes sharp convexities and the thin constrictions removed by shrinking.

The dual of rounding is filleting. S filleted by r , denoted $F_r(S)$, is $S \uparrow_r \downarrow_r$ (Fig. 2 (b)). $F_r(S)$ is the complement of the union of all B_r that lie in $\neg S$. Like $S \downarrow_r$, $F_r(S)$ is closed and the minimum curvature of $\partial i F_r(S)$ is greater than or equal to $-1/r$. $F_r(S) \supseteq S$, and $F_r(S) - S$ contains the portions of $\neg S$ in the concave corners of S as well as the gaps of S filled by growing.

2.5. Mortar, core, and anticore

The mortar $M_r(S)$ is $F_r(S) - R_r(S)$. Being the union of $F_r(S) - S$ and $S - R_r(S)$, $M_r(S)$ contains all of the details associated with S , including sharp convexities and concavities as well as the thin parts of both S and $\neg S$. Away from these details, the mortar is identical to ∂S .

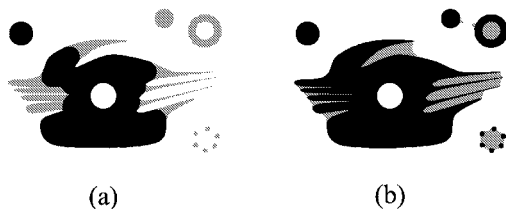


Fig. 2. (a) $R_r(S)$ is shown in black, and the material it removes from S is shown in gray. (b) S is shown in black, and the material $F_r(S)$ adds to S is shown in gray.

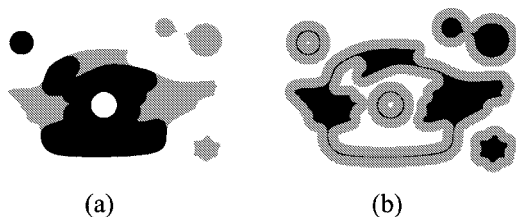


Fig. 3. (a) The core (black,) mortar (gray,) and anticore (white) for the set S shown in Fig. 1. (b) $M_r(S)$ (black) and $Z_r(S)$ (black and gray) for the set S shown in Fig. 1.

The mortar is the complement of the union of all open balls disjoint from ∂S . Consequently, the minimum curvature of $\partial iM_r(S)$ is greater than $-1/r$. As with $S \downarrow_r$ and $F_r(S)$, any point on the lower-dimensional portion of the mortar can be approached on either side by a ball of radius r , so the magnitudes of its curvatures are less than $1/r$. In addition, the lower-dimensional portion is G^1 -continuous with $\partial iM_r(S)$ along their intersection.

We use the mortar as a tolerance zone, and to emphasize that they are not affected by tightening we refer to $R_r(S)$ as the core and $-F_r(S)$ as the anticore of S . Together, the core, mortar, and anticore decompose space into three mutually disjoint sets, where the mortar is closed and the core and anticore are open (Fig. 3 (a)).

We can compare the mortar to the tolerance zone $Z_r(S)$ defined earlier as $(\partial S) \uparrow_r$ (Fig. 3 (b)). The mortar can be expressed as $F_r(\partial S)$, which is $(\partial S) \uparrow_r \downarrow_r$. Consequently, $M_r(S) = Z_r(S) \downarrow_r$, which entails that $M_r(S) \subset Z_r(S)$. Because it is the set of points at a distance of r or more from $-Z_r(S)$, the mortar is usually a small subset of $Z_r(S)$.

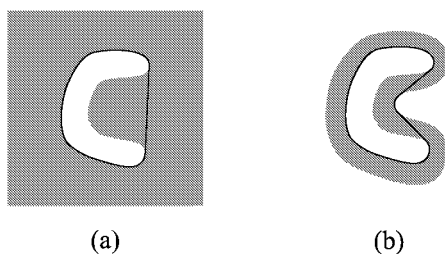


Fig. 4. The set X is shown in gray while the curve C that is tight with respect to X is shown in black. (a) X is the plane minus a hole. (b) X is a crescent-shaped annulus.

2.6. Two-dimensional r -tightenings

In two dimensions, r -tightenings are bounded by minimum-length loops lying in the mortar. We define a simple closed curve C lying in a closed set X as tight with respect to X if and only if it has locally minimal length. In other words, C is tight if and only if any small deformation of C that keeps it within X would increase its length.

First observe that if C is tight with respect to X , it must surround one or more components of $\neg X$. Otherwise, it could be continuously shrunk to a single point. Consider, for instance, the case where X is the plane minus a hole (Fig. 4 (a)). Then the only curve that is tight with respect to X is the convex hull of the hole. Next consider the case where X is a crescent-shaped annulus (Fig. 4 (b)). There is still only one tight curve C . It contains the portion of the convex hull of the hole that lies along the boundary of the hole, but it is now pushed inwards by a concave piece of the outer loop of ∂X . The parts of C that lie along ∂X are smoothly connected by straight line segments that lie in the interior of X .

Note that X may contain lower-dimensional portions, which are given by $\partial X - \partial iX$. C may then be split into three disjoint subsets: $C \cap iX$, $C \cap \partial iX$, and $C \cap (\partial X - \partial iX)$. If C is tight with respect to X , $C \cap iX$ must consist of line segments, $C \cap \partial iX$ must consist of concave (or flat) portions of ∂iX , and components from those two sets must be connected to each other with G^1 continuity.

A two-dimensional r -tightening of S is a set T such that $R_r(S) \subseteq T \subseteq F_r(S)$ and the boundary of T consists of disjoint simple closed curves that are tight with respect to $M_r(S)$. As discussed previously, the magnitude of the radius of curvature of both the lower-dimensional portions of the mortar and the concave parts of $\partial iM_r(S)$ is greater than or equal to r , and the lower-dimensional portions connect to $\partial iM_r(S)$ with G^1 continuity. It follows that the boundary of an r -tightening is r -smooth.

If the core and anticore of S each consist of a single connected component, the r -tightening is unique, and its boundary is the shortest loop around $R_r(S)$ lying in $F_r(S)$. When the core and anticore have more complex topologies, there may

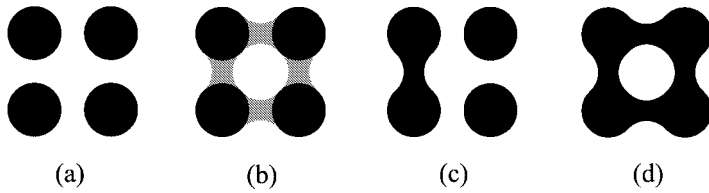


Fig. 5. (a) The input set S . (b) The core (black,) mortar (gray,) and anticore (white) of S . (c) An r -tightening of S with three components. (d) An r -tightening of S with a hole.

be several different r -tightenings, which may have holes and multiple connected components (Fig. 5).

2.7. Three-dimensional r -tightenings

When we move to three dimensions, the set-theoretic definition of the mortar is unchanged; it is dimension-independent. Instead of being bounded by curves of locally minimal length, however, three-dimensional r -tightenings are bounded by surfaces of locally minimal area. We say that a shell (a watertight surface without boundary consisting of a single connected component) is tight with respect to a closed set X that contains it when any small deformation of the shell that keeps it in X would increase its surface area. An r -tightening of a three-dimensional set S is then a set T such that $R_r(S) \subseteq T \subseteq F_r(S)$ and ∂T consists of disjoint simple closed shells that are tight with respect to $M_r(S)$.

Although we had a bound on the absolute value of the curvature of two-dimensional tightenings, three-dimensional r -tightenings are only guaranteed to have bounded mean curvature. To justify this claim, we again split ∂T into three parts. Because ∂T is of locally minimal area, $\partial T \cap iM_r(S)$ must have zero mean curvature, while ∂T can coincide with $\partial iM_r(S)$ only where the mean curvature of $\partial iM_r(S)$ is not positive. As in the two-dimensional case, components of $\partial T \cap iM_r(S)$ and $\partial T \cap \partial iM_r(S)$ must be smoothly joined. Coupled with the facts that the minimum curvature of $\partial iM_r(S)$ is greater than $-1/r$, the magnitudes of the principal curvatures of the lower-dimensional $\partial M_r(S) - \partial iM_r(S)$ are less than $1/r$, and $\partial M_r(S) - \partial iM_r(S)$ is G^1 -continuous with $\partial iM_r(S)$, this implies that the magnitude of the mean curvature of an r -tightening is less than or equal to $1/r$. We cannot exclude the possibility, however, that the absolute values of the principal curvatures of $\partial T \cap iM_r(S)$ might be greater than $1/r$.

In summary, the mortar $M_r(S)$ is a small subset of the tolerance zone $Z_r(S)$ often used to constrain shape simplification. Tightening smooths S by changing it only in $M_r(S)$. Consequently, it exactly preserves the portions of ∂S away from details, where the mortar is lower-dimensional. Among all possible shapes that contain the core and exclude the anticore, tightenings are those with locally minimal

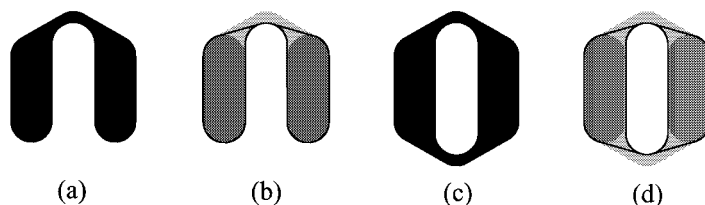


Fig. 6. (a) The input set S . (b) The core of S is shown in dark gray, the mortar is light gray, and the loop obtained by reducing the length of ∂S while preserving its topology is shown in black. The set bounded by the loop has two cusps and is not an r -tightening. The tightening in this case would be equal to the core. (c) and (d) A similar example where two loops overlap.

boundary. We have avoided a definition specifying a unique r -tightening, because factors such as which topology is preferable or whether globally-minimal area is desired may depend on the application.

3. Implementation

3.1. Two-dimensional implementation

In this section we present an algorithm for computing an approximation of an r -tightening for an input set represented as a binary image. The algorithm is easy to implement and applicable to a variety of datasets commonly used in image processing and morphology. One could also envision computing r -tightenings exactly, for instance, for shapes bounded by a combination of line segments and circular arcs.

Our strategy for computing an r -tightening of S is to continuously deform the bounding loops of S in a way that reduces their length while constraining them to lie in the mortar. If we preserve the loops' topology as we tighten them, some loops may eventually overlap themselves, or distinct loops can overlap (Fig. 6). Then, although the loops individually are smooth, the set they bound can have cusps. We can, however, arrive at a set with a smooth boundary if we let the loops split, merge, or disappear as they evolve. We opt to perform the evolution in a level set framework, which eliminates the need to explicitly handle these topology changes.

In our implementation, the evolving set S' corresponds to a function $\Phi(t, x, y)$ represented at a given time by floating point values on a pixel grid. We consider pixels with negative values to be in S' and pixels with positive values to be out. We move the boundary of S' by increasing or decreasing values that are close to those defining the zero level set.

We use curvature flow for the boundary motion. In the continuous case, curvature flow moves a point on a curve along its normal at a speed proportional to the curvature at that point. This dampens oscillations of the curve by pushing convex boundary in and concave boundary out. In the absence of constraints, this process eventually deforms a curve to a circle, which ultimately collapses to a point.²

Curvature flow is flow along the gradient of arclength (and mean curvature flow in three dimensions is flow along the gradient of surface area). Intuitively, a curve's arclength, as a function of the position of a point on the curve, changes most rapidly when the point moves normal to the curve, and the rate of change is proportional to the curvature at that point. This property of curvature flow makes it well-suited to finding curves of locally minimal length.

Before starting the flow, we compute the core and the anticore using a combination of growing and shrinking operations. To grow or shrink a set, we threshold its signed distance transform, which we obtain using Danielsson's approximate vector propagation algorithm.³ This algorithm runs in time linear in the number of pixels. It computes a vector offset from each pixel in the input set to the closest pixel in its complement, and from each pixel in the complement to the closest pixel in the set. It propagates these vectors outward from boundary pixels by sweeping left and right across each scanline, visiting scanlines first from top to bottom and then from bottom to top. It determines the vector at the current pixel in the sweep from the vectors at its neighbors. Although this procedure is not exact, the maximum error is bounded by 0.29 times the pixel grid's edge length.

Our implementation of curvature flow broadly follows the description in Ref. [4]. To start the flow, we initialize Φ to be the signed distance to the boundary of the core of S using vector propagation. At each iteration, we compute $\Phi_t = -F |\nabla\Phi|$ at each pixel, where F is the velocity of the level set. We set the velocity to zero in the core and anticore, so they remain respectively inside and outside the evolving set. F is equal to the curvature in the mortar, where the curvature at a point is equal to the divergence of the unit normal to the level set passing through that point. Because the normal to the level set is the gradient of the level set function, the formula for the curvature κ is $\nabla \cdot (\nabla\Phi / |\nabla\Phi|)$. We use finite differences to compute the components and magnitude of the gradient in one pass, and then apply finite differences to the normalized gradient components in a second pass to obtain the curvature.

Once we obtain the values of Φ_t , we update the level set function values using

$$\Phi(t + \Delta t, x, y) = \Phi(t, x, y) + \Delta t \cdot \Phi_t(t, x, y). \quad (1)$$

We set the value of Δt to be inversely proportional to the maximum value of F at time t , so that the level set crosses at most one pixel each iteration.

We accelerate this computation using a narrow band technique.⁴ We traverse the image and build a list of pixels adjacent to pixels of the opposite sign. We then use breadth-first search to expand this list to include all pixels within a chessboard distance of three pixels from a sign change. We confine updates of Φ to pixels in this band as long as the zero level set remains within the band. When it leaves the band we traverse the image again to construct a new band. Overall this achieves an order of magnitude speedup.

We further accelerate convergence by down-sampling the image representation of the core by a factor of two until r is less than twice a pixel size. We perform



Fig. 7. (a) The input set. The image is 1102 x 832 pixels. (b) The tightening of the input computed with a radius of 40 pixels.

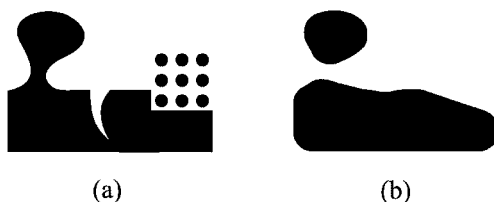


Fig. 8. (a) The input set. The image is 1050 x 825 pixels. (b) The tightening of the input computed with a radius of 75 pixels.

the flow at this coarse resolution, then iteratively up-sample by a factor of two and re-perform the flow. In principle there are cases where the mortar can be arbitrarily thick compared to r and the zero level set must consequently cross a large number of pixels at the coarsest level of resolution. However, we have found that 100 iterations at the coarsest level suffices when the mortar's thickness is less than $5r$, and at most 100 iterations are also required to smooth the boundary after up-sampling.

Figure 7 (b) shows the tightening of the set from Fig. 1 computed using this technique with a radius of 40 pixels. It took 2 seconds to compute the morphological operations for the 1102 x 832 pixel input image and another 5 seconds to perform 100 iterations of curvature flow at each of five levels of resolution on a dual 2 GHz PowerPC G5 with 2 GB RAM.

Figure 8 (a) shows a different input image and Fig. 8 (b) shows its tightening with a radius of 75 pixels. The image size for the second example is 1050 x 825 pixels, and it required 3 seconds for the morphological operations and 4 seconds for the multiscale flow. Because the radius is larger, we employed six levels of resolution instead of five, but because the image at the coarsest resolution is small the flow at that resolution adds little to the execution time.

For a fixed number of iterations of curvature flow, the asymptotic complexity of the entire tightening procedure is linear in the number of pixels in the input. It

does not depend on the radius.

3.2. Three-dimensional implementation

We also use constrained level set curvature flow to compute three-dimensional r -tightenings. To avoid faceting and terracing artifacts, we must represent the constraints with sub-voxel precision, rather than as binary volumes. Our input is a triangle mesh bounding a solid S . We begin by computing the signed distance to the mesh at each point on a grid. We then extract one isosurface at the value r , representing $S \uparrow_r$, and another isosurface at $-r$, representing $S \downarrow_r$. We compute signed distance transforms for both of these solids and again extract offset surfaces from each to obtain $F_r(S)$ and $R_r(S)$. Finally we compute distance transforms of $F_r(S)$ and $R_r(S)$. In total, we perform five transforms.

We then initialize the level set function to the signed distance to S and perform curvature flow, but rather than set the velocity to zero outside the mortar, we constrain the level set function value at each point to be less than the signed distance to $R_r(S)$ and greater than the signed distance to $F_r(S)$. This guarantees that the zero level set remains in the mortar.

Figure 9 shows an input mesh consisting of 829,604 triangles and its tightening, computed on a $128 \times 128 \times 279$ voxel grid with a radius of 8 voxels. The time required by the five distance transforms varied from 1 hr 16 min 32 sec to 1 hr 24 min 49 sec, with a total time of 6 hr 47 min 28 sec on the same platform as used in the 2D results. After the transforms were computed, we performed 300 iterations of curvature flow, which required 17 min 7 sec.

Figure 10 shows a second example mesh consisting of 1,031,984 triangles and its tightening, computed on $211 \times 145 \times 287$ voxel grid with a radius of 12 voxels. The time required by the distance transforms ranged from 2 hr 11 min 52 sec to 2 hr 49 min 26 sec, with a total time of 13 hr 6 min 5 sec. It then took 32 min 17 sec to perform 300 iterations of flow.

Distance transforms dominate our execution times in 3D, and it may be possible to improve our transform times. Our current times were obtained using an adaptation of vector propagation to compute distances to a surface. The results in Ref. [5] suggest an order-of-magnitude speedup could be achieved, provided the tessellation error the authors discuss does not introduce artifacts into our offset surfaces. The largest examples they present are smaller than our meshes, but if their technique or a variation of it scales reasonably well we estimate it could reduce our distance transform times from a couple of hours to several minutes each. Although the impact may not be as dramatic, level set optimizations such as the narrow band technique could accelerate the 3D flow.

A limitation of using isosurface extraction with distance transforms on a regular grid to compute morphological operations is that sharp and thin features arise in offsetting that may be captured poorly, particularly if the grid is coarse. This can in turn lead to artifacts in the tightening. Figure 11 (a) shows the sawtooth pattern

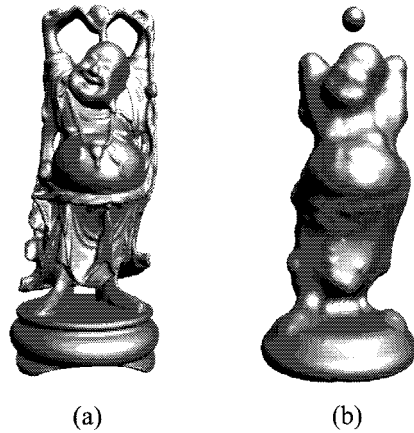


Fig. 9. (a) An input mesh consisting of 829,604 triangles. (b) Its tightening, computed on a 128 x 128 x 279 voxel grid with a radius of 8 voxels.

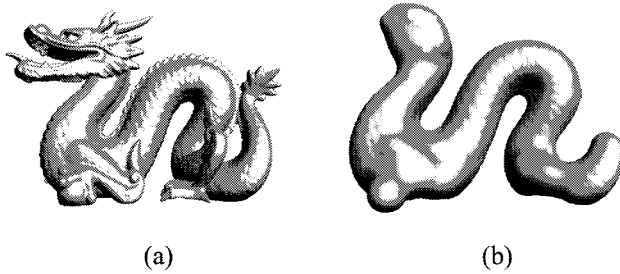


Fig. 10. (a) An input mesh consisting of 1,031,984 triangles. (b) Its tightening, computed on a 211 x 145 x 287 voxel grid with a radius of 12 voxels.

that appears in the base of the Buddha model when it is shrunk with a radius of 4 voxels on a 67 x 67 x 142 grid. This yields a bumpy edge when the model is grown again to obtain the rounding (b), and the bumps are retained in the tightening (c). It may be possible to address this issue by detecting thin edges and either refining the sampling in their vicinity, which would require adapting the distance transform, or repairing the isosurface after it is extracted. Alternatively, we could adopt a completely different approach to computing offsets, such as directly tessellating the bounding surfaces of the union of grown triangles.

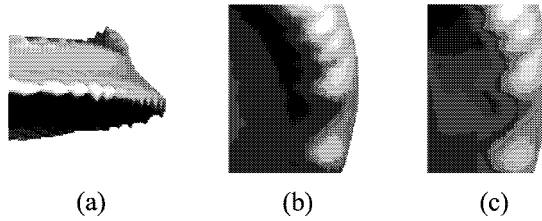


Fig. 11. (a) A sawtooth pattern that appears along the edge of the base of the Buddha model when it is shrunk with a radius of 4 voxels on a $67 \times 67 \times 142$ voxel grid. (b) When the shrunk model is grown to obtain the rounded model, this results in a bumpy edge. (c) The bumps are retained in the tightening.

4. Existence

We conjecture that for every set S and radius r , an r -tightening of S exists. As a lemma we conjecture that if $R_r(S) \subseteq T \subseteq F_r(S)$ and the boundary of T is not tight with respect to $M_r(S)$, there exists a T' such that $R_r(S) \subseteq T' \subseteq F_r(S)$ and $\partial T'$ is manifold and the measure of $\partial T'$ is less than the measure of ∂T . Intuitively, if we can monotonically decrease the perimeter or area of a set as long as its boundary is not tight, then because the measure of its boundary cannot decrease without bound we eventually converge to a boundary whose measure is locally minimal, that is, the boundary of a tight set. We use level-set curvature flow as a means for reducing perimeter or area and obtaining a manifold boundary, although we believe that other mechanisms are possible.

5. Discussion in the Context of Prior Art

5.1. Mathematical morphology

Growing, shrinking, rounding, and filleting are examples of operations developed in the field of mathematical morphology. There is a large literature on mathematical morphology; classic texts include Refs. 6 and 7. In this literature, growing and shrinking are referred to as the Minkowski sum and difference with an open ball, or as dilation and erosion with a ball. Rounding and filleting are referred to as opening and closing with a ball. The open ball B_r is called the structuring element.

5.2. Morphological simplification

Tightening is closely related to combinations of rounding and filleting such as $R_r(F_r(S))$ and $F_r(R_r(S))$.⁸ Like tightening, these filters only change the shape in the mortar. They also tend to produce shapes that are r -regular,⁹ where a set S is r -regular if $R_r(S) = F_r(S)$, so that the mortar is empty and both S and $\neg S$ are a union of B_r . Regularity implies smoothness, so $R_r \circ F_r$ and $F_r \circ R_r$ also tend to produce shapes that are r -smooth. However, $R_r(F_r(S))$ and $F_r(R_r(S))$ are not

guaranteed to be regular or smooth. They may contain irregularities such as cusps and constrictions. In Ref. [10] we proved that for some S there is no r -regular shape that only differs from S in the mortar. However, with tightening we can construct a shape that is r -smooth in two dimensions. This shape has no cusps, although it may still contain constrictions.

In Ref. [10], we also pointed out that $R_r \circ F_r$ and $F_r \circ R_r$ are not self-dual, where a filter Ψ is self-dual iff $\neg\Psi(S) = \Psi(\neg S)$. This lack of self-duality means that they do not treat positive and negative space symmetrically, and in particular $R_r \circ F_r$ is biased to add material in regions containing details, while $F_r \circ R_r$ is biased to remove it. This led us to introduce the self-dual Mason filter,¹⁰ which replaces each connected component of the mortar's interior with the result of applying $R_r \circ F_r$ or $F_r \circ R_r$, depending on which yields a smaller symmetric difference with the input set. Because r -tightenings are not unique, there may be several different functions that compute r -tightenings, not all of which are necessarily self-dual. However, it is possible to define a self-dual tightening function. For instance, the function that maps S to the result of performing constrained level-set curvature flow with S as the initial condition is self-dual. If we use $\neg S$ as the input the core and the anticore change roles but the mortar and the front propagation within the mortar remain the same.

We consider this filter to be an improvement on prior art in morphological simplification because we regard self-duality and smoothness (in 2D) or bounded mean curvature (in 3D) as desirable properties. However, the filter may fail to eliminate constrictions that would be removed by a combination of rounding and filleting, which can be viewed as undesirable if the objective is to produce the most nearly regular set possible. We can overcome this limitation by using Mason to generate the initial condition for the flow.

Another alternative, which is also self-dual, is to change the constraints. Rather than perform the flow in $M_r(S)$, we can perform it in $M_r(\text{Mason}(S))$. The mortar of $\text{Mason}(S)$ is a subset of $M_r(S)$ whose boundary has a minimum curvature greater than or equal to $-1/r$. The result is by definition not an r -tightening of S , but it has bounded mean curvature and only differs from S in the mortar. Using $M_r(\text{Mason}(S))$ may be of interest because it can make curvature flow converge faster. $M_r(\text{Mason}(S))$ is typically significantly thinner than $M_r(S)$, so that the front crosses fewer pixels during its evolution.

5.3. Medial axis

Morphological operations can be formulated in terms of the medial axis. The medial axis $MA(S)$ is the set of all points that have more than one closest point on ∂S ; note that $MA(S)$ may contain points in $\neg S$ as well as S . With each point q of $MA(S)$ we can associate the distance $r(q)$ from q to ∂S and the ball $B_{r(q)}$ with center q and radius $r(q)$. We can then express S as the union of $B_{r(q)}$ for all $q \in MA(S) \cap S$. We can similarly express the core as the union of $B_{r(q)}$ for all $q \in MA(S) \cap S$ such

that $r(q) > r$. We can obtain the anticore from $MA(S) \cap \neg S$, while the mortar is the complement of the union of $B_{r(q)}$ for $r(q) > r$. Reconstruction of a shape from a subset of the medial axis is reminiscent of the simplification algorithm used in Ref. [11].

The local feature size $lfs(p)$ is defined for $p \in \partial S$ as the distance between p and $MA(S)$.¹² Regularity can be defined in terms of local feature size: a set S is r -regular if and only if the minimum value of $lfs(p)$ is greater than or equal to r .

5.4. α -Shapes

We can obtain a discrete approximation of the mortar using α -shapes.¹³ Suppose we have a sufficiently dense set of samples P from ∂S and a Delaunay triangulation D of P . With each k -dimensional simplex σ in D we can associate the smallest circumsphere b_σ of the points defining the simplex and its radius r_σ . The simplex σ is in the α -complex of S if and only if either b_σ is empty and $r_\sigma < \alpha$ or σ bounds a $k + 1$ dimensional simplex in the α -complex of S . The α -shape of P is then the union of all the simplices in the α -complex of P . Like the mortar of S , the α -shape of P can contain lower-dimensional portions corresponding to the regular parts of the boundary of S . A full dimensional simplex τ with $r_\tau > \alpha$ corresponds to a portion of the core or anticore.

5.5. Topology simplification

Tightening can simplify topology by merging nearby shapes and filling small holes in S and $\neg S$. This leads us to compare it to other topology-simplifying techniques. Many techniques seek to reduce the number of elements in a polygonal mesh; several authors have found that allowing topology changes enables more dramatic reductions in vertex count. Some approaches use topology-modifying decimation operations, such as vertex merges.^{14,15,16} Other approaches, however, perform topology simplification separate from decimation. They include using standard morphological operations,^{17,18} volumetric lowpass filtering,¹⁹ and explicit handle removal.^{20,21} We discussed the standard morphological operations in Sec. 5.2. Although, like tightening, low-pass filtering can round off convexities and fill in concavities, it does not necessarily decrease curvature. Furthermore, it tends to erode the shape along regular portions that tightening would preserve. Finally, the explicit handle reduction techniques are not concerned with smoothness or regularity.

5.6. Fairing

Tightening can be considered a surface fairing method. We can broadly categorize fairing techniques into optimization-based approaches and polygon mesh smoothing. The optimization-based approaches appear in the CAD/CAM literature on problems such as ship hull and car body design.^{22,23,24,25} The objective of these methods is to deform a surface composed of curved patches to minimize an energy

function. A variety of functions have been proposed. Some directly penalize high curvature by, for instance, penalizing integral square curvature. Like tightening, minimizing such a function may decrease maximum curvature, although it would not guarantee a curvature bound. Other functions, such as those that penalize variation in curvature, have a more tenuous link to tightening.

There is also an extensive computer graphics literature on polygon mesh smoothing, which is often used to eliminate acquisition artifacts from laser range scanning or isosurface extraction. Mesh smoothing techniques can be variously related to robust statistics,^{26,27} signal processing,^{28,29} and partial differential equations.^{30,31} The goals of tightening can be distinguished from those of techniques such as bilateral filtering and anisotropic diffusion that seek to eliminate noise while preserving sharp features and consequently might not reduce maximum curvature. Our use of curve evolution differs from those of the remaining techniques in that our objective is to shrink the shape as quickly as possible while depending on the mortar to ensure that this shrinking both eliminates high curvature and preserves low curvature parts of the boundary. Other approaches, which do not employ a tolerance zone, seek to selectively affect high-curvature boundary while minimizing or correcting for the effect of evolution on the gross shape. For instance, the use of curvature flow in Ref. [30] relies on the fact that it dampens small-scale, high-curvature oscillations more quickly than it affects the slowly varying parts of the shape. The authors propose to correct for the global shrinking that occurs by rescaling the surface to preserve volume, which can displace the flatter contours that the mortar preserves. Similarly, signal processing approaches are designed to attenuate high frequencies while retaining low frequencies. They were in fact originally developed to eliminate the shrinking associated with Laplacian smoothing.²⁸ To the extent that frequency can be equated with curvature, tightening can be considered a filter with a precise cutoff: it perfectly preserves regular parts of the curve while completely removing parts where the magnitude of the mean curvature is greater than $1/r$.

5.7. Relative convex hull

The idea of finding a set bounded by the shortest path around a core set inside an outer set appears in the work by Sklansky *et al.* on the minimum perimeter polygon.³² This set was later termed the relative convex hull or geodesic convex hull. If the core and outer sets are both simple polygons with $O(n)$ sides, the relative convex hull can be computed using path planning in a simple polygon, which can be performed in $O(n)$ time,³³ while if the core is a set of $O(n)$ points and the outer set is a polygon of $O(n)$ sides the relative convex hull requires $O(n \cdot \log n)$ time.³⁴ The link between convexity and minimum boundary is specific to two dimensions; the minimum surface enclosing a solid is not necessarily convex.

6. Conclusion

We have defined two- and three-dimensional r -tightenings for input sets of arbitrary topology. Every shape smoothing technique reflects a compromise between fidelity to the input shape and the smoothness of the output shape. Tightening offers strong, precise guarantees on both fidelity and smoothness. It exactly preserves the r -regular portions of the input, confining changes to the shape to the details in the mortar. Because $M_r(S) = Z_r(S) \downarrow_r$, this is a significantly stronger guarantee than requiring that each point on the boundary of the tightening be within a distance r of the input. An r -tightening is nevertheless guaranteed to be r -smooth (in 2D) or to have mean curvature with a magnitude bounded by $1/r$ (in 3D.) There may be several tightenings of an input shape, which offers the opportunity to select the tightening that best suits a given application. Our definitions of r -tightenings and the mortar are independent of shape representation, and it may be possible to compute tightenings using a variety of algorithms. We have described how to compute approximate r -tightenings for sets represented as binary images or triangle meshes using constrained, level-set curvature flow.

Acknowledgment

This research was supported by a DARPA/NSF CARGO grant number 0138420.

References

1. J. Rossignac and A. Requicha, Offsetting operations in solid modeling, *Comput. Aided Geom. Des.* **3** (1986) 129–148.
2. M. Grayson, The heat equation shrinks embedded plane curves to round points, *J. Diff. Geom.* **26** (1987) 285–314.
3. P. Danielsson, Euclidean distance mapping, *Comput. Graph. Image Process.* **14** (1980) 227–248.
4. J. Sethian, *Level Set Methods and Fast Marching Methods* (Cambridge University Press, New York, 1999).
5. A. Sud, M. Otaduy and D. Manocha, DiFi: Fast 3D distance field computation using graphics hardware, *Comput. Graph. Forum* **23** (2004) 557–566.
6. H. Heijmans, *Morphological Image Operators* (Academic Press, Boston, 1994).
7. J. Serra, *Image Analysis and Mathematical Morphology* (Academic Press, London, 1982).
8. J. Rossignac, Blending and offsetting solid models, Ph.D. Thesis, University of Rochester (1985).
9. D. Attali, r -Regular shape reconstruction from unorganized points, *Proc. 13th Ann. Symp. Computational Geometry*, Nice, France (Jun. 1997) pp. 248–253.
10. J. Williams and J. Rossignac, Mason: Morphological simplification, *Graph. Models* **67** (2005) 285–305.
11. R. Tam and W. Heidrich, Shape simplification based on the medial axis transform, *Proc. IEEE Visualization 2003*, Seattle, WA (Oct. 2003) pp. 481–488.
12. N. Amenta and M. Bern, Surface reconstruction by Voronoi filtering, *Discr. Comput. Geom.* **22** (1999) 481–504.

13. H. Edelsbrunner and E. Mucke, Three-dimensional alpha shapes, *ACM Trans. Graphics* **13** (1994) 43–72.
14. J. Rossignac and P. Borrel, Multi-resolution 3D approximations for rendering complex scenes, in *Geometric Modeling in Computer Graphics*, eds. B. Falcidieno and T. Kunii (Springer-Verlag, New York, 1993) 455–465.
15. C. Erickson and D. Manocha, GAPS: General and automatic polygonal simplification, *Proc. 1999 Symp. Interactive 3D Graphics*, Atlanta, GA (Apr. 1999), pp. 79–88.
16. J. Popovic and H. Hoppe, Progressive simplicial complexes, *Proc. ACM SIGGRAPH 97*, Los Angeles, CA (Aug. 1997), pp. 217–224.
17. J. El-Sana and A. Varshney, Topology simplification for polygonal virtual environments, *IEEE Trans. Visual. Comput. Graph.* **4** (1998) 133–144.
18. F. Nooruddin and G. Turk, Simplification and repair of polygonal models using volumetric techniques, *IEEE Trans. Visual. Comput. Graph.* **9** (2003) 191–205.
19. T. He, L. Hong, A. Varshney and S. Wang, Controlled topology simplification, *IEEE Trans. Visual. Comput. Graph.* **2** (1996) 171–183.
20. A. Szymczak and J. Vanderhyde, Extraction of topologically simple isosurfaces from volume datasets, *Proc. 14th IEEE Visual. 2003*, Seattle, WA (Oct. 2003), pp. 67–74.
21. Z. Wood, H. Hoppe, M. Desbrun and P. Schroder, Removing excess topology from isosurfaces, *ACM Trans. Graph.* **23** (2004) 190–208.
22. G. Celniker and D. Gossard, Deformable curve and surface finite elements for free-form shape design, *Proc. ACM SIGGRAPH 91*, Las Vegas, NV (Jul. 1991), pp. 257–266.
23. S. Hahmann and S. Konz, Fairing bi-cubic B-spline surfaces using simulated annealing, in *Curves and Surfaces with Applications in CAGD*, eds. A. Le Mehaute, C. Rabut and L. Schumaker (Vanderbilt University Press, Nashville, TN, 1997), pp. 159–168.
24. H. Moreton and C. Sequin, Functional optimization for fair surface design, *Proc. ACM SIGGRAPH 92*, Chicago, IL (Jul. 1992), pp. 167–176.
25. G. Westgard and H. Nowacki, Construction of fair surfaces over irregular meshes, in *Proc. 6th ACM Symp. Solid Modeling and Applications*, Ann Arbor, MI (Jun. 2001), pp. 88–98.
26. S. Fleischman, I. Drori and D. Cohen-Or, Bilateral mesh denoising, in *Proc. ACM SIGGRAPH 03*, San Diego, CA (Jul. 2003), pp. 950–953.
27. T. Jones, F. Durand and M. Desbrun, Non-iterative, feature-preserving mesh-smoothing, in *Proc. ACM SIGGRAPH 03*, San Diego, CA (Jul. 2003), pp. 943–949.
28. G. Taubin, Curve and surface smoothing without shrinkage, in *Proc. 5th Int. Conf. Computer Vision*, Cambridge, MA (Jun. 1995), pp. 852–857.
29. H. Zhang and E. Fiume, Butterworth filtering and implicit fairing of irregular meshes, *Proc. 11th Pacific Conf. Computer Graphics and Applications*, Canmore, Canada (Oct. 2003), pp. 502–506.
30. C. Bajaj and G. Xu, Anisotropic diffusion of surfaces and functions on surfaces, *ACM Trans. Graph.* **22** (2003) 4–32.
31. M. Desbrun, M. Meyer, P. Schroder and A. Barr, Implicit fairing of irregular meshes using diffusion and curvature flow, *Proc. ACM SIGGRAPH 99*, Los Angeles, CA (Aug. 1999), pp. 317–324.
32. J. Sklansky, R. Chazin and B. Hansen, Minimum perimeter polygons of digitized silhouettes, *IEEE Trans. Comput.* **21** (1972) 260–268.
33. J. Mitchell, Shortest paths and networks, in *Handbook of Discrete and Computational Geometry*, eds. J. Goodman and J. O’Rourke (CRC Press, Boca Raton, FL, 2004), pp. 607–641.
34. G. Toussaint, Computing geodesic properties inside a simple polygon, *Revue D’Intelligence Artif.* **3** (1989) 9–42.