

Structural MAT: Clean and Scalable Medial Axis Simplification via Explicit Surface Correspondence

PENGFEI WANG, Shandong University, China

SHUANGMIN CHEN, Qingdao University of Science and Technology, China and Shandong Key Laboratory of Deep Sea Equipment Intelligent Networking, China

DONGMING YAN, Institute of Automation, Chinese Academy of Sciences, China

YING HE, Nanyang Technological University, Singapore

SHIQING XIN*, Shandong University, China

CHANGHE TU*, Shandong University, China

WENPING WANG, Texas A&M University, United States of America

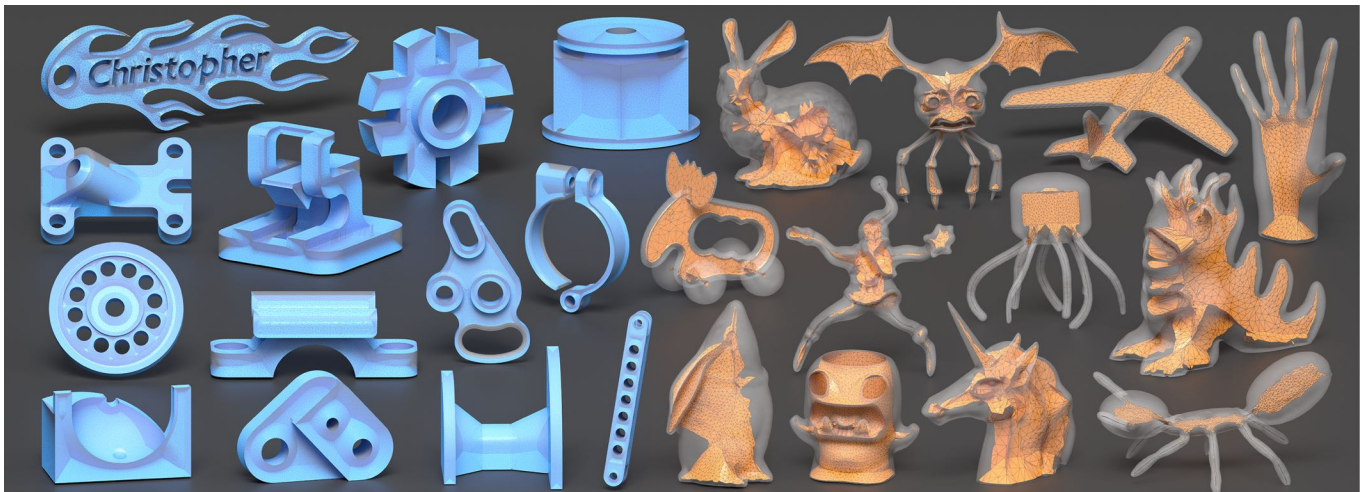


Fig. 1. Representative medial axis results computed by our method on CAD models with sharp features and organic models with smooth surfaces. The surface-guided simplification framework ensures clean medial boundaries, accurate feature alignment, and high mesh quality across diverse geometric inputs.

The Medial Axis Transform (MAT) is a complete shape descriptor capable of reconstructing the geometry of the original domain. A high-quality MAT should not only facilitate high-fidelity reconstruction but also capture structural features—for instance, by aligning the MAT boundary with the locus of rolling ball centers within fillet regions. However, computing such an ideal MAT remains a significant challenge, particularly when the input is a discrete triangle mesh.

*Corresponding authors.

Authors' Contact Information: Pengfei Wang, Shandong University, Qingdao, China, pengfei1998@foxmail.com; Shuangmin Chen, School of Information and Technology, Qingdao University of Science and Technology, Qingdao, China and Shandong Key Laboratory of Deep Sea Equipment Intelligent Networking, Qingdao, China, csmqq@163.com; Dongming Yan, Institute of Automation, Chinese Academy of Sciences, Beijing, China, yandongming@gmail.com; Ying He, Nanyang Technological University, Singapore, Singapore, yhe@ntu.edu.sg; Shiqing Xin, Shandong University, Qingdao, China, xinshiqing@sdu.edu.cn; Changhe Tu, Shandong University, Qingdao, China, chtu@sdu.edu.cn; Wenping Wang, Texas A&M University, College Station, United States of America, wenping@tamu.edu.

© 2026 Copyright held by the owner/author(s). Publication rights licensed to ACM. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Graphics*, <https://doi.org/10.1145/nnnnnnn.nnnnnnn>.

In this paper, we follow the established technical pipeline of initializing the MAT via a 3D Voronoi diagram of surface samples and subsequently simplifying the Voronoi structure through a QEM-like scheme. Our key insight is to explicitly track the correspondence between MAT vertices and surface regions throughout the progressive simplification process, ensuring that the resulting MAT triangles accurately reflect the intrinsic symmetries between surface patches. We translate these geometric requirements into a suite of priority control strategies that govern the sequencing of edge collapses.

Through extensive evaluation against state-of-the-art MAT algorithms, we validate the strong performance of our approach regarding runtime efficiency, structural alignment, boundary regularity, triangle quality, and robustness to noise. Our resulting MATs remain highly expressive for both articulated shapes and CAD models, even under extreme simplification—effectively capturing the global structure of complex geometries with only a few hundred vertices. Finally, we showcase the utility of our approach through two potential applications: capturing the locus of rolling ball centers within fillet regions, a structural capability not previously demonstrated in the existing literature, and surface extraction from unsigned distance fields, where the medial axis of the ϵ -isosurface naturally yields a clean single-layer result.

Source code is available at <https://github.com/sssosome/structural-mat>.

CCS Concepts: • **Computing methodologies** → **Mesh models**; *Shape analysis*.

ACM Reference Format:

Pengfei Wang, Shuangmin Chen, Dongming Yan, Ying He, Shiqing Xin, Changhe Tu, and Wenping Wang. 2026. Structural MAT: Clean and Scalable Medial Axis Simplification via Explicit Surface Correspondence. *ACM Trans. Graph.* 45, 4, Article 164 (May 2026), 19 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 Introduction

Mathematically, the medial axis of a bounded shape is defined as the set of points with at least two nearest points on the boundary—or equivalently, as the locus of centers of maximally inscribed spheres. Together with their associated radii, these points form the Medial Axis Transform (MAT), a compact and complete descriptor that encodes a shape’s topological structure and local thickness. Due to its powerful geometric abstraction capabilities, the MAT is a cornerstone of shape understanding, supporting a wide range of downstream applications including shape analysis [Dou et al. 2020; Fu et al. 2022; Hu et al. 2019; Noma et al. 2024], shape decomposition [Lin et al. 2020; Zhou et al. 2015], pose analysis [Dou et al. 2023; Yang et al. 2021], and animation [Lan et al. 2020, 2021].

The quality of a medial mesh representation is typically evaluated by two standards: its ability to accurately reconstruct the original surface and its capacity to effectively inherit structural features. Furthermore, the triangle quality of the MAT mesh itself is a critical factor for downstream stability. Most prior research has prioritized reconstruction accuracy while paying less attention to structural alignment. For instance, Q-MAT [Li et al. 2016] employs progressive edge collapses where the collapse priority is based primarily on the minimization of reconstruction loss. However, as simplification proceeds, the connection between a MAT vertex and the underlying surface symmetry weakens, leading to structural drift.

While recent works such as MATFP [Wang et al. 2022], MAT-Topo [Wang et al. 2024b], and MATStruct [Wang et al. 2025c] have improved geometric fidelity and topological correctness for CAD models through feature-aware optimization, challenges remain in balancing computational efficiency, mesh quality, and geometric accuracy. Despite these considerations, existing methods struggle to produce regularized, zigzag-free boundaries. Figure 2 illustrates a toy model with round fillets; it is evident that previous methods fail to align the MAT boundary with the locus of rolling ball centers within fillet regions.

In this paper, we shift the focus toward the structural symmetry of the original surface to produce a structurally reliable MAT. We adopt the established technical pipeline of initializing the MAT via the Voronoi diagram of dense surface samples followed by progressive simplification via edge collapses. Our key insight is to maintain an explicit mapping between MAT vertices and surface regions throughout the simplification process. Initially, the Voronoi diagram constructs the 3D medial mesh, which naturally induces a partition on the surface manifold (known as the Restricted Voronoi Diagram). This establishes a rigorous 2D-3D correspondence: each 3D medial vertex is intrinsically linked to several specific surface patches. When a MAT edge is collapsed, the associated surface regions of the two endpoints are inherited by the new vertex. The optimal

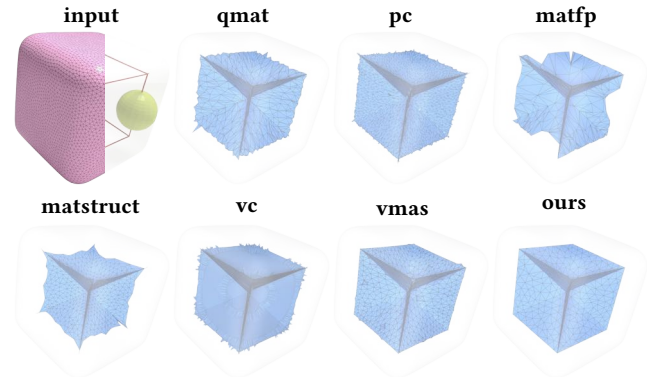


Fig. 2. Medial axis computation on a filleted cube model. The true medial axis boundaries should align with the centerlines traced by the rolling sphere centers in the filleted regions. Our approach captures this structure, demonstrating its capability to faithfully sense and represent intrinsic structural features. Notably, VMAS [Huang et al. 2024] is evaluated with the correction step disabled due to optimization instability.

position of this new vertex is then determined by these associated surface regions, ensuring that the MAT remains anchored to the shape’s bilateral symmetry. Simultaneously, we integrate a topological laplacian smoothing metric into the vertex positioning logic to promote a well-conditioned MAT mesh.

By leveraging these strategies, our surface-guided simplification framework maintains explicit correspondence between medial vertices and their associated surface regions throughout the entire process, enabling all decisions to directly reference the original geometry. This persistent correspondence allows our approach to produce clean, structure-aware medial axes with regularized, zigzag-free boundaries for both organic shapes and complex CAD models across varying levels of simplification; see Figure 1.

- We develop a structural MAT framework that tracks the correspondence between medial vertices and surface regions, ensuring the MAT captures intrinsic surface symmetries and maintains structural alignment.
- We propose a positioning strategy that jointly optimizes for geometric accuracy and triangulation quality, resulting in robust MAT meshes with clean boundaries across both organic and man-made geometries, with resilience to moderate levels of noise.
- We demonstrate the utility of our structural MAT in identifying the locus of rolling ball centers within fillet regions. Extracting this locus directly from discrete mesh representations is a significant challenge that, to our best knowledge, has not been addressed in the existing literature.

2 Related Work

The Medial Axis Transform (MAT) is a fundamental shape descriptor with extensive applications in shape approximation [Petrov et al. 2024; Yang et al. 2020], simplification [Yan et al. 2016], and analysis [Lin et al. 2021]. Over the past decades, numerous methods have been proposed to approximate the 3D medial axis, each navigating

the trade-offs between geometric accuracy, topological correctness, and computational efficiency. For comprehensive surveys on medial axis computation, we refer readers to [Siddiqi and Pizer 2008; Tagliasacchi et al. 2016]. In this section, we review representative approaches most relevant to our work, categorized by their underlying methodology.

2.1 Voronoi-based Medial Axis Computation

The Voronoi diagram [Lévy 2026; Lévy and Bonneel 2013; Wang et al. 2025a,b; Yan et al. 2009] is a primary tool for medial axis computation due to its spatial partitioning properties and inherent link to the medial axis. In 2D, if the boundary curve satisfies ϵ -sampling conditions, the subset of the Voronoi diagram contained within the shape converges to the medial axis [Brandt 1994]. However, this property does not directly extend to 3D due to the presence of “slivers”—Voronoi vertices positioned infinitesimally close to the surface that produce redundant, unstable medial branches [Amenta et al. 2001].

To address this, various filtering techniques have been developed. *Angle-based filtering methods* [Attali and Montanvert 1996; Dey and Zhao 2004] prune the Voronoi diagram using geometric criteria, though they frequently fail to guarantee topological preservation. The λ -*medial axis* [Chazal and Lieutier 2005, 2008] discards medial spheres with radii smaller than a threshold λ ; while this preserves homotopy, it tends to over-smooth fine geometric features. Alternatively, the *Scale Axis Transform (SAT)* [Giesen et al. 2009; Miklos et al. 2010] applies multiplicative scaling to medial spheres to remove unstable spikes while retaining small features by filtering balls that are “devoured” by their neighbors. Beyond filtering, *Power Crust* [Amenta et al. 2001] utilizes the weighted Voronoi diagram of “poles” to establish connectivity, forming a power shape that converges to the medial axis as sampling density increases.

2.2 Voxel-based Methods

Rather than relying on discrete surface samples, voxel-based approaches discretize the target shape into a grid and extract medial subsets based on distance fields, such as Euclidean [Hesselink and Roerdink 2008; Rumpf and Telea 2002] or chamfer distances [Pudney 1998]. *Voxel Cores (VC)* [Yan et al. 2018] provides rigorous sampling conditions and computes the medial mesh as the dual of the interior Delaunay triangulation. While these methods offer theoretical guarantees of topological equivalence, they often face a trade-off between geometric precision and computational overhead: high accuracy requires extremely fine voxel resolutions, leading to significant memory consumption.

2.3 PD-based and Patch-based Optimization

Recent advances have moved toward using Power Diagrams (PD) for feature-aware MAT computation. *MATFP* [Wang et al. 2022] employs surface Restricted Power Diagrams (RPD) to optimize Voronoi ball positions, ensuring they remain tangential to the boundary. It notably preserves external sharp edges by placing zero-radius spheres on non-smooth regions. *MATTopo* [Wang et al. 2024b] further enhances topological robustness using volumetric RPDs, though the

required tetrahedron slicing operations are computationally intensive. Both methods attempt to preserve internal features (seams and junctions) by inserting feature spheres when structural deficiencies are detected, which can occasionally lead to numerical instability or densely clustered primitives.

To improve mesh quality, *MATStruct* [Wang et al. 2025c] introduces a particle-based energy optimization framework. However, the iterative recomputation of power diagrams within the optimization loop remains a major computational bottleneck. Similarly, patch-based methods [Wang et al. 2025a] segment models into patches to compute generalized Voronoi diagrams. While effective for simple CAD models, these methods struggle when patch boundaries are ambiguous or when features between adjacent patches are not well-defined.

2.4 Simplification and Sparse Approximation

Given the density of raw medial extractions, several approaches focus on generating sparse representations. *Q-MAT* [Li et al. 2016] employs iterative simplification via edge collapses guided by a quadric error metric. While flexible, Q-MAT suffers from significant “reference drift”; because each step relies on the previous approximation rather than the original surface, geometric fidelity weakens over time, leading to irregular boundaries.

Conversely, *skeletal point selection methods* [Dou et al. 2022; Wang et al. 2024a] take a constructive approach by selecting a target number of skeletal points and inferring connectivity. *VMAS* [Huang et al. 2024] adopts a coarse-to-fine strategy by minimizing a hybrid metric. However, VMAS does not explicitly account for feature preservation—neither external sharp edges nor internal seams. Additionally, it can exhibit oscillatory behavior or instability when the sphere count exceeds a few hundred, preventing convergence to a clean, sparse representation. Our method addresses these limitations by leveraging the persistent correspondence between the 3D Voronoi diagram and the surface RVD, ensuring structural alignment even at extreme simplification levels.

3 Problem Formulation and Overview

3.1 Problem Statement

The Medial Axis Transform (MAT), as a compact and complete descriptor that encodes a shape’s topological structure and local thickness, plays a fundamental role in geometric representation. Given a continuous or discrete boundary surface \mathcal{S} , the objective is to compute a medial structure \mathcal{M} , typically represented as a discrete mesh consisting of vertices, edges, and faces. In engineering applications, this discrete outcome often serves as the basis for further transformation into quadrilateral representations or B-reps. Despite significant progress in this domain, several intrinsic challenges persist, which are further amplified when the input is a discrete mesh representation rather than a smooth manifold.

Requirements for a High-Quality MAT. To characterize a high-quality MAT, we identify a set of often-conflicting requirements that a robust algorithm should satisfy:

- (1) *Geometric Fidelity:* The ability to accurately reconstruct the original surface \mathcal{S} with minimal geometric deviation.

- (2) *Topological Inheritance*: The preservation of the topological properties, such as genus and homotopy type, of the original surface.
- (3) *Structural Alignment and Boundary Regularity*: The capacity to effectively capture intrinsic structural features—such as aligning boundaries with fillet centerlines—while maintaining clean, zigzag-free boundaries.
- (4) *Mesh Quality*: Ensuring well-conditioned triangle elements to avoid normal flips, near-zero area triangles, and other forms of mesh degeneracy.
- (5) *Conciseness and Scalability*: The ability to remain geometrically representative even at extremely low triangle counts, achieving ultra-low-poly representations.
- (6) *Computational Efficiency*: Optimized runtime performance suitable for processing large-scale or complex geometric inputs.

Most prior research has prioritized reconstruction accuracy while paying less attention to structural alignment and boundary regularity. In this paper, our algorithm focuses on achieving high geometric fidelity and computational efficiency through a correspondence-preserving simplification framework. As a natural consequence of maintaining explicit medial-to-surface correspondence throughout simplification, our method also improves structural alignment and boundary regularity, while balancing the competing demands of mesh quality and conciseness. Regarding topological inheritance, we incorporate a heuristic protection strategy that helps reduce the risk of unintended topological changes during simplification, though strict preservation is not guaranteed for models of arbitrarily high genus.

3.2 Overview

A common pipeline for MAT computation begins with a set of surface samples $\{s_i\}_{i=1}^n$ extracted from \mathcal{S} , where \mathcal{S} is either a smooth surface or a discrete representation. This is followed by a pruning and simplification process of the Voronoi diagram \mathcal{V} constructed from $\{s_i\}_{i=1}^n$. Since the medial axis of a bounded shape lies strictly inside its boundary, only the inner portion of \mathcal{V} —comprising Voronoi vertices, edges, and faces enclosed by \mathcal{S} —is retained as the initial medial representation; all exterior components are discarded. While previous methods often treat the simplification purely in the volumetric domain, we explicitly leverage the intrinsic geometric link between the dual structures: each 3D Voronoi cell naturally induces a corresponding region on the surface manifold (the Restricted Voronoi Diagram cell). In our framework, we strictly enforce this correspondence, treating each surface sample s_i as a proxy for this specific local surface region (see Figure 3(a)).

In a general configuration, each Voronoi vertex $v_j \in \mathcal{V}$ is equidistant to three boundary samples in 2D or four surface samples in 3D. Consequently, v_j can be regarded as being jointly determined by three boundary segments in 2D or four surface regions in 3D, as illustrated in Figure 3(b,c). We define the set

$$\text{Atlas}(v_j) = \{\text{Cell}(s_i) \mid s_i \text{ is a contributing site for } v_j\} \quad (1)$$

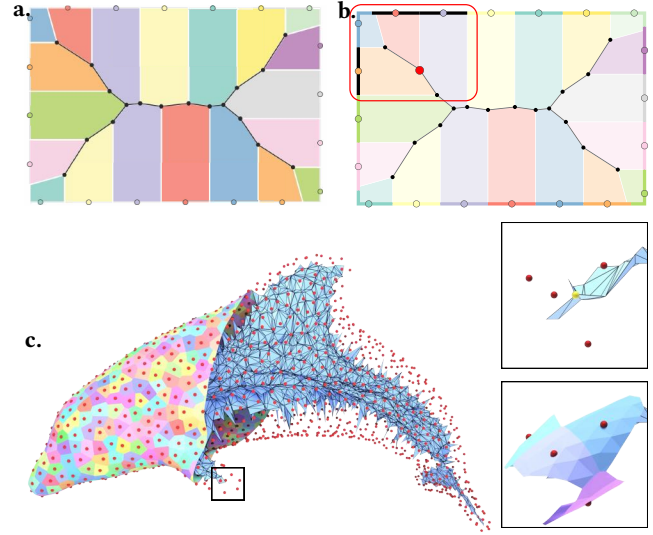


Fig. 3. The geometric correspondence between the boundary surface and medial axis vertices. (a) 2D Construction: Surface samples generate the Voronoi diagram (MAT) and simultaneously induce the RVD on the boundary. (b) Local Correspondence: As highlighted, a single medial vertex is defined by three equidistant surface samples. We define the union of the three RVD cells belonging to these generating samples as the Atlas of the vertex, denoted as $\text{Atlas}(v_j)$. (c) 3D Duality: In our 3D framework, this concept generalizes: each medial vertex is defined by four surface samples, and its $\text{Atlas}(v_j)$ comprises the four corresponding surface cells.

to denote the collection of surface regions associated with a Voronoi vertex v_j . This set represents the fundamental geometric correspondence between a medial-axis primitive (i.e., a mesh vertex of the current medial-axis surface) and the boundary manifold.

This intrinsic duality inspires our framework to progressively simplify the MAT while explicitly maintaining this correspondence throughout the entire pipeline. Specifically, when an edge e is collapsed to form a new vertex v , we infer the optimal position of v based on the aggregated surface regions inherited from its two endpoints. Furthermore, we incorporate constraints for triangle quality and feature line alignment when determining the position of v . An overview of our pipeline is illustrated in Figure 4.

4 Method

Our method takes a triangulated surface mesh as input and produces a simplified, high-quality medial mesh through surface-guided optimization. The algorithm begins by sampling the input surface and computing dual Voronoi structures: the inner portion of a 3D Voronoi diagram in volumetric space, which serves as the initial medial representation, and a Restricted Voronoi Diagram (RVD) on the surface manifold, which provides geometric guidance throughout simplification. We then progressively refine this initial structure through iterative edge collapses, with each operation strictly anchored to the original input surface via the established correspondence.

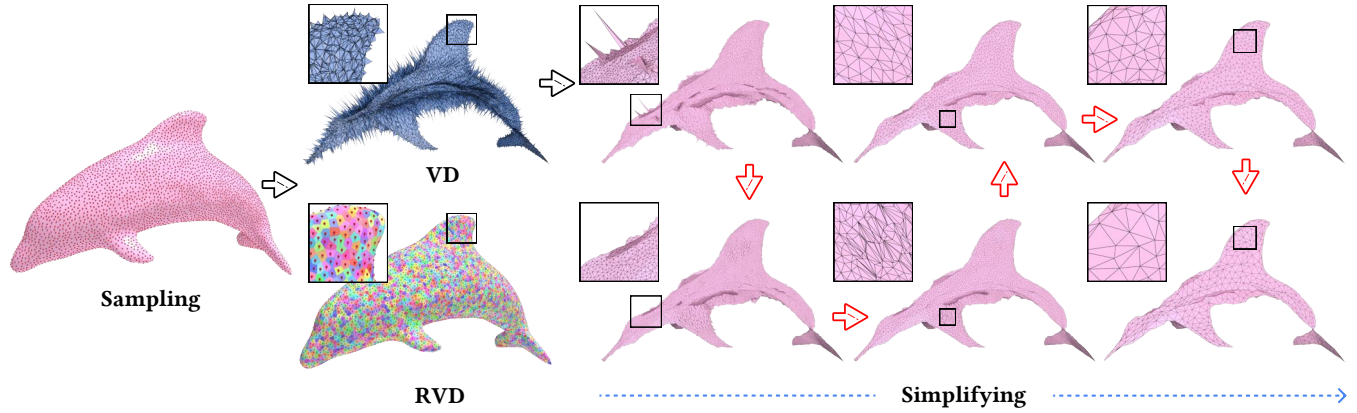


Fig. 4. Pipeline overview. We begin by sampling the input surface and computing the 3D Voronoi diagram along with the Restricted Voronoi Diagram (RVD) on the surface. Each vertex in the 3D Voronoi diagram is assigned a set of surface regions, establishing a correspondence between the medial vertex and the mesh surface. This correspondence is inherited and updated throughout the simplification process.

The remainder of this section is organized as follows. Section 4.1 describes the construction of the dual Voronoi structures and introduces a medial face filtering mechanism designed to eliminate spurious faces near concave features. Section 4.2 details the optimization framework for edge collapse, where we determine the optimal position of new vertices by minimizing a composite energy functional that balances surface fidelity with mesh quality. Section 4.3 presents our edge prioritization strategy, which ranks collapse operations based on a unified metric combining geometric error, element quality, and structural stability. Finally, Section 4.4 outlines the specialized handling for CAD models, ensuring that sharp features and corners are accurately preserved throughout the simplification process.

4.1 Initialization

Surface Sampling and Voronoi Construction. We begin by performing blue noise sampling on the input surface \mathcal{S} to obtain a dense set of surface samples, typically ranging from 10K to 100K points. These samples serve as the generating sites for a 3D Voronoi diagram \mathcal{V} . The Restricted Voronoi Diagram (RVD) is defined as the restriction of \mathcal{V} to the surface \mathcal{S} , where each RVD cell is given by:

$$\text{Cell}(s) = \mathcal{V}(s) \cap \mathcal{S}, \quad (2)$$

where $\mathcal{V}(s)$ is the Voronoi cell of s in \mathbb{R}^3 . This yields a partition of \mathcal{S} into surface regions, each associated with a unique generating sample.

In a general configuration, each Voronoi vertex v in \mathcal{V} is equidistant to exactly four surface samples, denoted as the set S_v . This duality between Voronoi vertices and surface regions is the key structure we exploit. We define the *Atlas* of vertex v as the union of the RVD cells belonging to its four generating sites:

$$\text{Atlas}(v) = \bigcup_{s \in S_v} \text{Cell}(s). \quad (3)$$

This establishes an explicit correspondence between each medial vertex v and a specific region of the input surface, effectively encoding the local bilateral symmetry of the shape (see Figure 3(c)).

Medial Face Filtering near Concave Features. In our dual Voronoi framework, each face in the 3D Voronoi diagram corresponds to two surface sample sites whose Voronoi cells share that interface. When both sites lie near concave feature lines, the medial face becomes geometrically ambiguous—it may represent a valid medial structure or merely a spurious artifact induced by the concave feature. Standard energy-based edge collapse cannot reliably distinguish and eliminate these questionable structures.

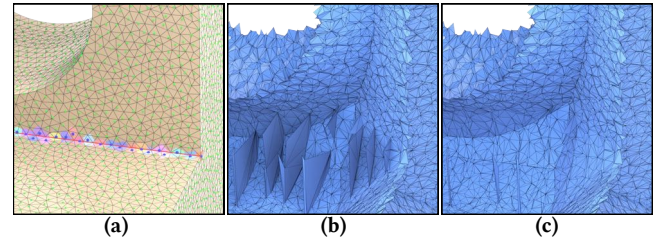


Fig. 5. Medial face filtering near concave features. (a) Concave features on the CAD model are marked with red lines. Cells intersecting these feature lines are highlighted, with their corresponding Voronoi sites marked in blue. (b) Initial medial structures often include spurious faces where both endpoints lie on the same concave feature line. (c) The medial structure after filtering removes faces that fail the midpoint test, resulting in a cleaner medial topology in concave regions.

We therefore apply a geometric test to filter such faces before simplification begins. This filtering is strictly restricted to medial faces generated by two sites whose respective Restricted Voronoi cells both intersect concave feature lines. For such a medial face f corresponding to surface sites s_1 and s_2 , we evaluate the midpoint $m = (s_1 + s_2)/2$. Our heuristic follows from a simple geometric observation: if the face generated by s_1 and s_2 truly lies on the medial axis, then its midpoint m must also lie on the medial axis. Consequently, m should satisfy the empty ball property, meaning its distance to the surface is determined solely by s_1 and s_2 . We identify

and remove spurious faces that violate this condition by checking:

$$d_S(m) < \alpha \cdot \frac{\|s_1 - s_2\|}{2}, \quad (4)$$

where $d_S(m)$ is the distance from m to the closest surface point, and $\alpha = 0.7$. If this inequality holds, it implies that m fails to lie on the medial axis, and therefore, the face f itself cannot be part of the valid medial axis. This preprocessing removes spurious structures near concave features, enabling more reliable subsequent simplification. Figure 5 shows the medial structure in concave regions before and after this filtering step, demonstrating the removal of spurious faces.

4.2 Optimal Vertex Placement

The operation of collapsing a medial edge $e \triangleq v_1v_2$ into a single vertex v involves two primary considerations: the geometric fidelity of the reconstructed boundary surface and the mesh quality of the medial surface triangulation. To determine the optimal target position v_{new} for an edge collapse, we introduce a composite energy functional consisting of two terms:

$$E(e \rightarrow v) = E_{\text{Fidelity}}(e \rightarrow v) + \lambda E_{\text{Lap}}(e \rightarrow v), \quad (5)$$

where v represents the candidate position for the new vertex, E_{Fidelity} penalizes geometric deviation from the original boundary, and E_{Lap} promotes well-conditioned triangle elements. The parameter λ is a weighting coefficient that balances these requirements. The optimal position is defined as the minimizer of this functional:

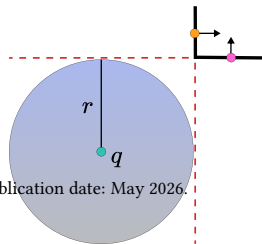
$$v_{\text{new}} = \arg \min_v E(e \rightarrow v). \quad (6)$$

We denote the aggregated surface regions as $\mathcal{A} = \text{Atlas}(v_1) \cup \text{Atlas}(v_2)$, inherited from the edge endpoints. Upon collapse, the target vertex v_{new} inherits this correspondence, meaning \mathcal{A} becomes the associated surface region for v_{new} and is continuously propagated through subsequent simplification steps. This ensures that every medial vertex remains linked to a specific patch of the original input surface.

Each Restricted Voronoi Diagram (RVD) cell within \mathcal{A} is classified as either *invaginated* or *regular*, depending on whether it intersects concave features. This classification is necessary because these two types of cells exhibit fundamentally different behaviors during optimization, requiring distinct energy formulations.

Invaginated vs. Regular Cells. For a point on a smooth surface region, the local tangent plane defined by the point and its normal generally provides a good approximation to the original surface in the point's neighborhood. Existing methods exploit this approximation by using the distance from a spatial location to the tangent plane as a proxy for the true distance to the surface. One of the most widely used techniques based on this principle is the Spherical Quadric Error Metric (SQEM) [Thiery et al. 2013].

However, this tangent plane approximation breaks down near concave features. When sample points lie close to concave (invaginated) feature lines, the distance to the tangent plane can deviate significantly from the true distance



to the surface. As illustrated in the inset, the signed distance from the medial sphere to the tangent plane fails to approximate the true distance to the boundary, rendering SQEM-based optimization unreliable in these regions.

To handle this issue, we classify each RVD cell based on its proximity to concave features. We detect concave features using dihedral angle thresholds: a surface edge is classified as concave if its dihedral angle exceeds $\pi + \phi$, where ϕ is a user-specified tolerance parameter. A cell Cell_i is marked as an *invaginated cell* if it intersects any concave feature line; otherwise, it is classified as a regular cell. Figure 5(a) illustrates the identified *invaginated cells* and their corresponding surface sites on a local region of a CAD model. This classification determines the appropriate optimization strategy for vertex placement during edge collapses.

Fidelity Term for Regular Cells. For a regular cell (one without concave features), we formulate its fidelity energy using the Spherical Quadric Error Metric (SQEM) framework [Thiery et al. 2013], which measures the distance between a medial sphere and surface elements. For a medial sphere $s = (v, r)$ with center v and radius r , the signed distance to a surface point p with normal n is:

$$d_{p,n}(v, r) = \mathbf{n}^\top (p - v) - r. \quad (7)$$

By defining the augmented state vector $\bar{v} = (v_x, v_y, v_z, r)^\top \in \mathbb{R}^4$, the squared distance can be expressed as a quadratic form:

$$d_{p,n}^2(v, r) = \bar{v}^\top A_p \bar{v} + \mathbf{b}_p^\top \bar{v} + c_p. \quad (8)$$

where A_p , b_p , and c_p are coefficients determined by point p and normal n . The fidelity energy for a single cell Cell_i is the integral of this metric over the cell region:

$$E_{\text{Fidelity}}(e \rightarrow v)|_{\text{Cell}_i} = \int_{p \in \text{Cell}_i} (\bar{v}^\top A_p \bar{v} + \mathbf{b}_p^\top \bar{v} + c_p) dA. \quad (9)$$

Since the surface region within each cell is composed of multiple planar polygons, the integral over the entire cell can be efficiently computed as the sum of area-weighted quadric terms from these individual facets. The quadratic coefficients for each cell can be pre-computed during initialization and accumulated through matrix additions during simplification.

Fidelity term for invaginated RVD cells. When an RVD cell Cell_i is invaginated, SQEM becomes problematic due to significant variations in surface normals within the patch. In such cases, we move away from normal-based distance and instead treat Cell_i as a geometric anchor for the medial sphere. We modify the formulation as follows:

$$E_{\text{Fidelity}}(e \rightarrow v)|_{\text{Cell}_i} = \int_{p \in \text{Cell}_i} (\|v - p\| - r)^2 dA. \quad (10)$$

To optimize for computational efficiency, we further relax this objective by using the cell's generating site s_i as a representative proxy

for the patch:

$$E_{\text{Fidelity}}(e \rightarrow v)|_{\text{Cell}_i} \approx \text{Area}(\text{Cell}_i) \cdot (\|v - s_i\| - r)^2, \quad (11)$$

where s_i is the site corresponding to the invaginated RVD cell Cell_i . This formulation effectively stabilizes vertex placement within invaginated regions.

Topological Laplacian smoothing term. When a medial edge $e \triangleq v_1 v_2$ is collapsed into a vertex v , let $\mathcal{N}(v_1, v_2)$ denote the union of the neighboring vertices of v_1 and v_2 (excluding v_1 and v_2 themselves). We incorporate topological (uniform) Laplacian smoothing term over the merged neighborhood to promote well-conditioned triangulation:

$$E_{\text{Lap}}(e \rightarrow v) = \frac{1}{|\mathcal{N}(v_1, v_2)|} \sum_{u \in \mathcal{N}(v_1, v_2)} \|v - u\|^2, \quad (12)$$

where $|\mathcal{N}(v_1, v_2)|$ denotes the cardinality of the neighbor set. Minimizing this term encourages the target vertex to be positioned near the centroid of its neighborhood, promoting local isotropy and the formation of near-equilateral triangles, which significantly improves the triangle quality and numerical stability of the simplified medial mesh for downstream applications.

Combined Optimization. The total energy combines fidelity contributions from all cells with a laplacian smoothing term:

$$E(e \rightarrow v) = \sum_{\text{Cell}_i \in \mathcal{A}} E_{\text{Fidelity}}(e \rightarrow v)|_{\text{Cell}_i} + \lambda E_{\text{Lap}}(e \rightarrow v) \quad (13)$$

The optimization strategy depends on whether \mathcal{A} contains *invaginated cells*. For regions without *invaginated cells*, both E_{Fidelity} (SQEM) and E_{Lap} are quadratic, reducing the optimization to solving a small linear system with a closed-form solution. For regions with invaginated cells, the square root term in Equation 11 makes the energy nonlinear, which we solve using L-BFGS optimization [Liu and Nocedal 1989]. In both the linear and nonlinear cases, the optimization variables are the four degrees of freedom of the candidate medial sphere, namely the center coordinates $v = (v_x, v_y, v_z)$ and the radius r . All other quantities—including the surface sites s_i , the surface points p with their normals \mathbf{n} , and the associated cell regions in \mathcal{A} —are treated as fixed constants inherited from the input surface at the moment of edge collapse, and are not updated during the optimization. In particular, \mathcal{A} is determined once by the union of the Atlases of the two endpoints and remains unchanged while the L-BFGS iterations refine (v, r) .

To ensure stability in the nonlinear case, we initialize v at the edge midpoint $\frac{v_1 + v_2}{2}$ and set r as the minimum distance from this midpoint to all surface sites in \mathcal{A} . This geometrically meaningful initialization improves convergence. The weight λ balances surface fidelity and mesh quality across both formulations.

The explicit gradient formulas with respect to (v, r) for both regular and invaginated cells are derived in Appendix A.

4.3 Priority for Edge Collapse

The energy functional defined in Section 4.2 serves a dual purpose. While its minimization yields the optimal target configuration, the resulting minimum energy value $E(e \rightarrow v_{\text{new}})$ provides a quantitative measure of the geometric deviation introduced by the collapse.

In a standard simplification framework, one would typically prioritize collapsing edges with the lowest reconstruction cost E to maximize fidelity.

However, for medial axis simplification, relying solely on geometric error is insufficient. A critical requirement is the effective removal of unstable branches ("spikes")—artifacts sensitive to boundary noise—before simplifying the stable structural components. To address this, we incorporate a structural stability measure into the prioritization logic.

To quantify structural stability, we directly adopt the stability measure from Q-MAT [Li et al. 2016]. For an edge $e = (v_1, v_2)$ with radii r_1 and r_2 , the stability ratio is defined as:

$$\text{Spike}(e) = \frac{\max\{0, \|v_1 - v_2\| - |r_1 - r_2|\}}{\|v_1 - v_2\|}. \quad (14)$$

This normalized metric ranges from 0 (representing a pure spike) to 1 (representing a stable edge).

To unify these two considerations into a single scalar for prioritization, we define a cost function:

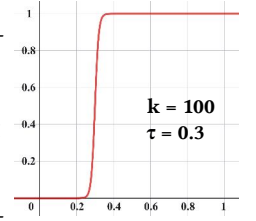
$$\text{Cost}(e) = E(e \rightarrow v_{\text{new}}) \cdot \Psi(\text{Spike}(e)), \quad (15)$$

where $\Psi(x)$ is a sigmoid-based weighting function:

$$\Psi(x) = \frac{1}{1 + e^{-k(x-\tau)}}. \quad (16)$$

In this formulation, τ acts as the stability threshold. We employ a large sharpness parameter (e.g., $k = 100$) to enforce a near-binary transition behavior:

- **Spike Removal:** For unstable edges ($\text{Spike}(e) < \tau$), Ψ vanishes, forcing their immediate removal regardless of geometric error.
- **Geometry Simplification:** For stable branches ($\text{Spike}(e) > \tau$), Ψ saturates to 1, ensuring the collapse order is dominated purely by geometric fidelity.



All candidate edges are maintained in a priority queue ordered by this weighted cost. The process iteratively collapses the least-cost edge and updates the neighborhood until the target complexity is reached.

Furthermore, to help preserve topological fidelity during the coarse simplification phase, we adopt a protection strategy similar to [Li et al. 2016]. Specifically, once the vertex count drops below 200, we enforce the *Link Condition* [Dey et al. 1998] prior to every edge collapse. This constraint explicitly identifies and forbids contractions that would alter the mesh topology—such as closing essential holes or collapsing tunnels—helping to reduce the risk of topological changes during simplification. It should be noted that this strategy is heuristic in nature: on models with highly complex topology, topological errors may still occur, particularly when the target vertex count is insufficient to faithfully represent the input genus.

4.4 Extension to CAD Models

CAD models contain numerous sharp features such as edges and corners. A fundamental characteristic of the MAT for such shapes

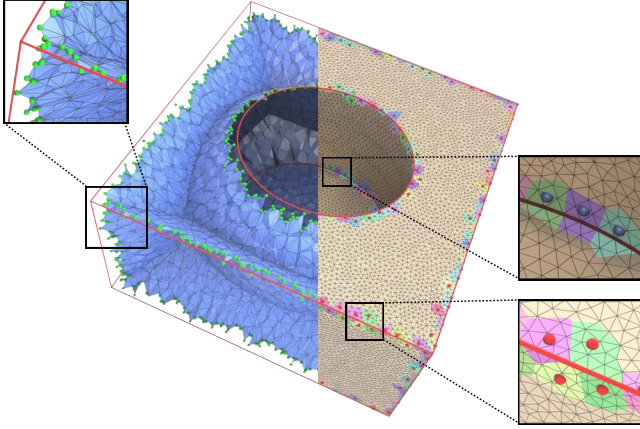


Fig. 6. Feature classification. Convex features and concave features on the model surface are marked in red and maroon, respectively. All Restricted Voronoi cells that intersect feature lines are highlighted, along with their corresponding sample sites (see right close-ups). Additionally, feature vertices are marked in green (left close-up).

is that the medial sheets extend outward and terminate precisely at convex feature lines, where the medial radius vanishes. To accurately capture and preserve these geometric details (which are often lost in smooth approximations), we introduce additional preprocessing and optimization strategies.

Feature Classification. Specifically, we define a surface edge as a *sharp feature* if its dihedral angle is less than $\pi - \phi$, where ϕ is a user-specified tolerance. Accordingly, any RVD cell intersecting such a sharp feature line is designated as a *feature cell*.

This cell-level characterization enables us to identify *feature vertices* in the medial mesh. A medial vertex v is classified as a *feature vertex* if its associated surface region $\text{Atlas}(v)$ contains at least one *feature cell* but is entirely free of *invaginated cells*. This criterion isolates stable, prominent corners or ridges for preservation, while explicitly excluding unstable geometry associated with invaginated regions. Figure 6 illustrates these categories and their geometric correspondence.

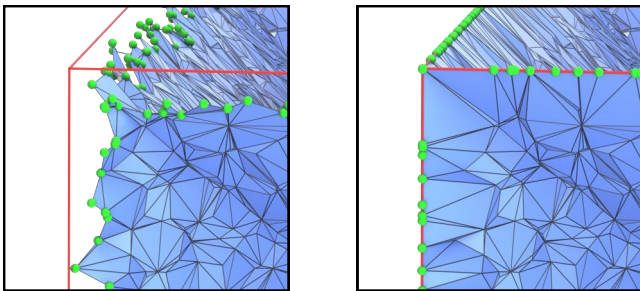


Fig. 7. Comparison of *feature vertices* before (left) and after (right) being snapped to convex feature lines. By enforcing $r = 0$, we prevent the medial boundary from drifting and ensure accurate capture of sharp CAD features.

Feature Snapping. Prior to simplification, we perform a preprocessing step to align *feature vertices* with convex boundary features. We adapt the energy functional defined in Equation 5 to a single-vertex setting. Crucially, we enforce a zero-radius constraint ($r = 0$) to drive the medial vertex onto the boundary. The optimization problem is formulated as:

$$v_{\text{snap}} = \arg \min_v \left(E_{\text{Fidelity}}(v, r = 0) \Big|_{\text{Atlas}(v)} + \lambda E_{\text{Lap}}(v) \Big|_{\mathcal{N}(v)} \right). \quad (17)$$

By fixing $r = 0$, the fidelity term reduces to minimizing the squared distance from v to the surface patch $\text{Atlas}(v)$, which is equivalent to the classical QEM formulation of [Garland and Heckbert 1997] in this setting. This naturally pulls v toward the sharp feature line where the medial radius vanishes. This effectively “snaps” the medial boundary to the CAD model’s sharp features, as illustrated in Figure 7, ensuring accurate capture of geometric details before simplification begins.

Feature Edge Collapse. To ensure that feature vertices remain strictly anchored to convex boundary features during simplification, we adopt a discrete candidate sampling strategy for edges incident to feature vertices (provided no invaginated RVD cells are involved). This choice is motivated by the need to prevent excessive drifting along feature lines and ensure numerical stability when the zero-radius constraint is enforced. Specifically, we select the target v from a finite set of candidates \mathcal{X} . If only one endpoint is a *feature vertex*, we restrict the candidate set to that vertex alone ($\mathcal{X} = \{v_{\text{feat}}\}$). If both are feature vertices, we consider both endpoints and their midpoint: $\mathcal{X} = \{v_1, v_2, (v_1 + v_2)/2\}$.

For each candidate $x \in \mathcal{X}$, we enforce a zero-radius constraint ($r = 0$) to preserve sharp features. The collapse cost is evaluated by adapting the general energy functional (Equation 15):

$$C_{\text{feature}}(e, x) = \left(E_{\text{Fidelity}}(x, 0) \Big|_{\mathcal{A}} + \lambda E_{\text{Lap}}(x) \Big|_{\mathcal{N}} \right) \cdot \Psi(\text{Spike}(e)), \quad (18)$$

where $\mathcal{A} = \text{Atlas}(v_1) \cup \text{Atlas}(v_2)$ is the aggregated surface region. The term $\Psi(\text{Spike}(e))$ is a weighting factor derived from the edge stability classification (as defined in Section 4.3), used to prioritize stable collapses. The candidate x^* minimizing this cost is selected as the collapse target, and the cost determines the edge’s priority in the simplification queue.

5 Evaluation

Experimental Setup. In this section, we present quantitative and qualitative evaluations of the proposed method. We implemented our algorithm in C++ and conducted experiments on two platforms: a Mac Mini with M4 chip and 16GB RAM served as our primary testing platform, while comparisons with methods requiring CUDA or Windows were performed on a system with Intel i7-14700K CPU, 64GB RAM, and NVIDIA RTX 4090 GPU. Our implementation uses SurfaceVoronoi [Xin et al. 2022] for RVD computation, CGAL [Jamin et al. 2025] for 3D Voronoi diagram construction, and FCPW [Sawhney 2021] for efficient point-to-mesh distance queries.

All input models were normalized to the unit cube $[0, 1]^3$ during testing. We set the algorithm parameters as follows: $\lambda = 6 \times 10^{-6}$ and $\tau = 0.025$. For surface sampling, we employed blue noise sampling. In comparative evaluations, we used 10K sample points and simplified the medial axis to 1K vertices for freeform models, and

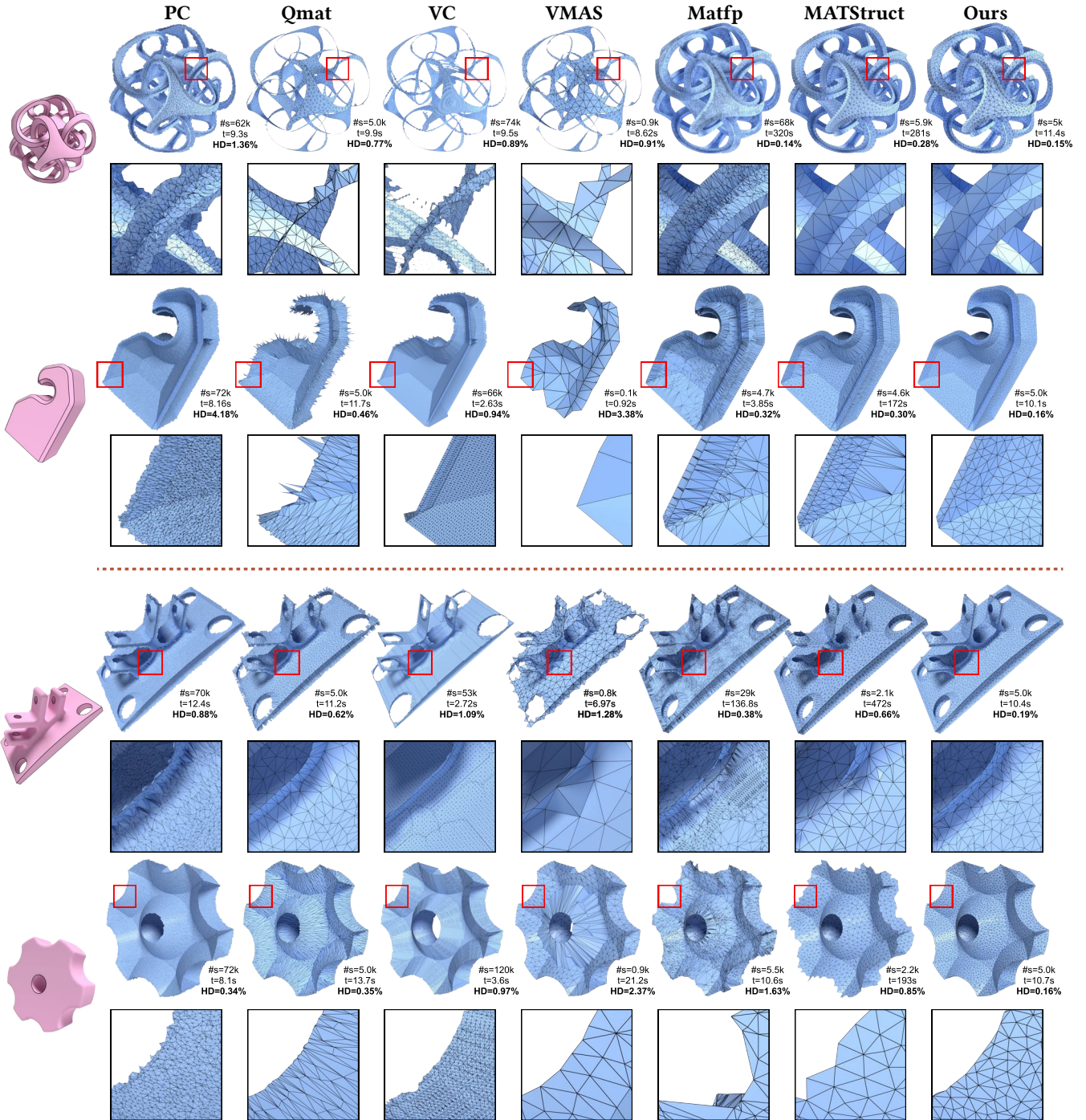


Fig. 8. Medial axis comparisons on representative CAD models with sharp features (top two rows) and regions with smooth transitions (bottom two rows). Results include computation time (t), sphere count ($\#s$), and the Hausdorff distance (HD) between the reconstructed surface and the original input. Our surface-guided approach accurately captures both sharp feature alignments and centerlines of smooth transitions.

50K sample points with simplification to 5K vertices for CAD models. Our test models were sourced from Thingi10K [Zhou and Jacobson 2016], the ABC Dataset [Koch et al. 2019], and [Huang

et al. 2024]. We compared our method against seven state-of-the-art approaches: Q-MAT [Li et al. 2016], VC (Voxel Cores) [Yan et al. 2018], PC (Power Crust) [Amenta et al. 2001], VMAS [Huang et al.

2024], MATFP [Wang et al. 2022], MATStruct [Wang et al. 2025c], and [Wang et al. 2025a]. Among these, MATStruct and VC were evaluated on the Windows platform, while all other methods were tested on the Mac platform.

Evaluation Metrics. We evaluate the reconstruction quality of the medial axis using bidirectional Hausdorff distance (HD) between the input surface and the reconstructed surface. All Hausdorff distances are reported as percentages relative to the diagonal length of the model’s bounding box. Surface reconstruction from the medial axis is performed using the Blender MAT addon [Song and Wang 2023]. In the comparative tables, $\#s$ denotes the number of medial spheres in the simplified medial mesh, and t represents the computation time in seconds.

Special Note on VMAS. Since the effective capacity of VMAS is inherently tied to the density of the initial sampling, its performance varies with the input configuration. In our experiments, the input meshes contain approximately 15K vertices on average. VMAS runs rapidly but does not terminate automatically, and the optimization may become unstable as the target sphere count increases. For VMAS, the reported time t reflects the duration until we manually stopped the optimization after observing stable convergence. All timings for VMAS are reported with artificial slow-down disabled for a fair benchmark.

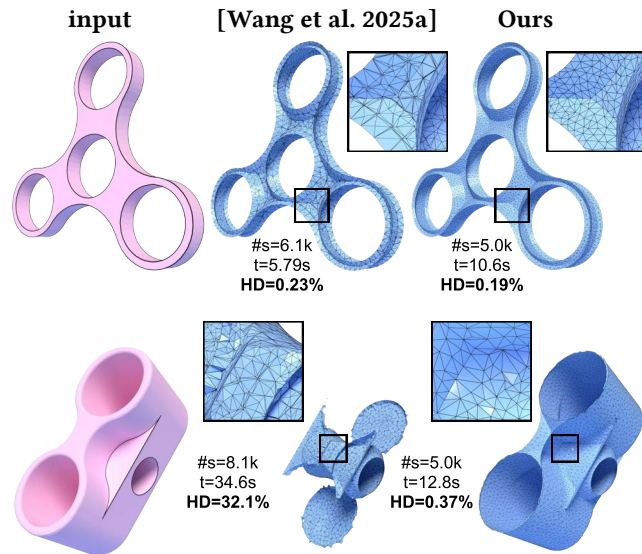


Fig. 9. Comparison with [Wang et al. 2025a] on CAD models. [Wang et al. 2025a] produces reasonable results when models can be cleanly segmented into patches (top models), but struggles or fails entirely when faced with models that resist straightforward patch decomposition (bottom models). Our method handles both categories uniformly without requiring segmentation preprocessing.

5.1 Comparisons on CAD Models

We evaluate our method on two representative categories of CAD models: models with sharp, well-defined features that can be detected via dihedral angle thresholds, and models containing smoothly

blended edges where features gradually transition without clear boundaries. While sharp features can be identified through preprocessing, smooth transitions present a more challenging test—the true medial axis should trace the centerline of the transitional region. An algorithm’s ability to accurately capture these centerlines reflects its capacity to faithfully represent the overall surface geometry.

Figure 8 shows comparative results on both categories, including computation time (t), number of medial spheres ($\#s$), and bidirectional Hausdorff distance (HD). Methods such as PC, Q-MAT, VC, and VMAS were not designed with CAD-specific handling in mind, and consequently struggle to capture sharp features—their results exhibit irregular, jagged boundaries that fail to align with the geometric features of the input models.

MATFP [Wang et al. 2022] and MATStruct [Wang et al. 2025c] represent state-of-the-art approaches specifically designed for CAD models. For models with sharp features, both methods demonstrate clear advantages over general-purpose approaches: their medial boundaries are better aligned with surface features, marking a significant step forward. However, both methods also exhibit certain limitations. MATFP produces meshes with suboptimal triangle quality, while MATStruct achieves better triangulation at the cost of occasionally missing narrow feature regions (see the first model in Figure 8). Additionally, MATStruct requires iterative power diagram recomputation, leading to slow performance and GPU dependency that limits its applicability. Both methods also exhibit limitations in capturing internal medial features—in some cases, the medial structure deviates from expected straight feature lines, appearing curved or irregular where geometric features should be linear (observe the second model in Figure 8). For models containing smooth feature transitions, the limited surface awareness of both methods hinders their ability to accurately trace the centerlines through these gradual changes, resulting in irregular medial boundaries that fail to capture the smooth transitional regions.

Figure 9 compares our method with [Wang et al. 2025a], which computes the medial axis by segmenting CAD models into patches and computing generalized Voronoi diagrams for each patch. For models amenable to clean segmentation, [Wang et al. 2025a] produces reasonable results, though with poor triangle quality. However, the method struggles or fails entirely when faced with models that resist straightforward patch decomposition.

Our method addresses these limitations through its surface-guided framework. The explicit correspondence between medial spheres and surface regions enables accurate feature alignment for sharp-featured models, while the continuous SQEM formulation provides the geometric fidelity needed to capture filleted edges. The result is a unified approach that handles both categories effectively without requiring specialized preprocessing or GPU acceleration.

5.2 Comparisons on Organic Models

We further evaluate our method against six competing approaches on multiple organic (freeform) models. Figure 10 shows comparative results on four representative examples, annotating computation time (t), bidirectional Hausdorff distance (HD), and the number of medial spheres ($\#s$) for each method.

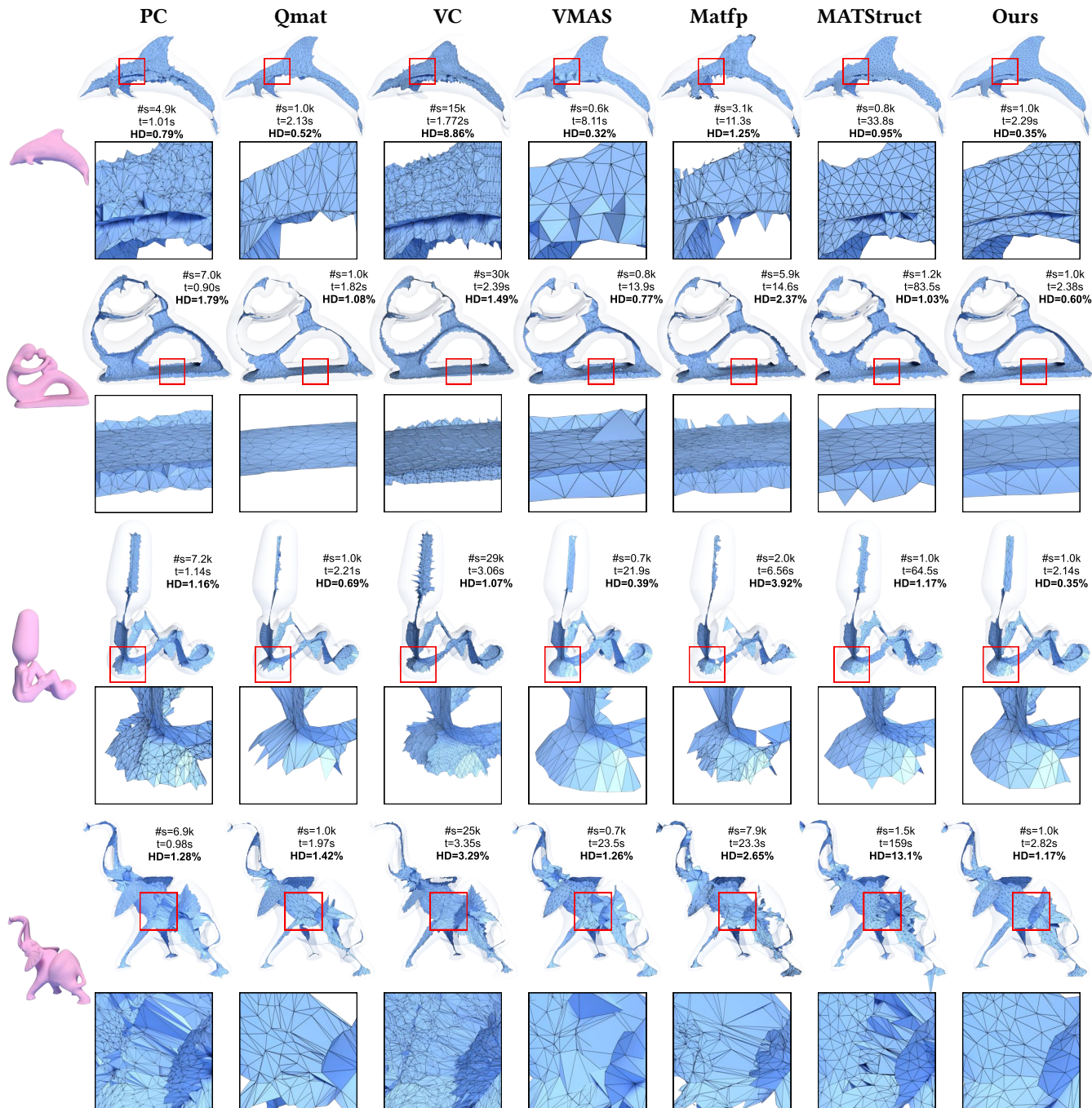










Fig. 10. Medial axis comparisons on four organic models, showing computation time (t), sphere count (#s), and the Hausdorff distance (HD) between the reconstructed surface and the original input.

PC generates medial axes containing spurious spike structures (see the first model). Q-MAT employs iterative simplification where each step relies on the previous iteration's result, causing progressive error accumulation that weakens surface fidelity and produces irregular, zigzag boundaries (observe the third model). VC loses

important structural details during computation, such as failing to preserve the dolphin's fin in the first model. VMAS demonstrates strong approximation quality with few spheres through iterative insertion and optimization. However, the method stalls as sphere

Table 1. Timing statistics for our method on representative models. For each model, we report the total computation time (in millisecond) and the time distribution across key stages: sampling, RVD computation, 3D Voronoi computation, queue initialization, and edge collapse simplification.

Models	# Samples	Time (ms)					Total
		Sampling	RVD	Voro3D	Init	Simp	
	10K	296.15	190.89	123.80	345.32	1337.9	2294.0
	10K	300.04	169.09	132.08	305.30	1470.8	2386.1
	10K	295.95	116.09	139.34	315.03	1277.3	2143.7
	10K	308.57	196.71	145.19	332.55	1822.3	2827.2
	50K	1737.5	724.93	1139.4	1331.7	6478.8	11444
	50K	1301.7	359.78	939.02	864.19	6680.3	10146
	50K	1533.3	445.71	832.24	970.97	6640.6	10435
	50K	1404.6	621.30	824.17	1040.0	6781.9	10679

insertion cannot proceed effectively at higher sphere counts. Additionally, VMAS establishes mesh connectivity based on the surface tessellation induced by medial spheres, which leads to structurally disordered results (models 1, 2, and 4). MATFP produces results with unwanted spike artifacts. MATStruct achieves high triangle quality but produces structurally incorrect results in some cases (models 1 and 4).

Our method demonstrates consistent results across all test cases, maintaining geometric fidelity and mesh quality while avoiding many of the artifacts observed in competing approaches.

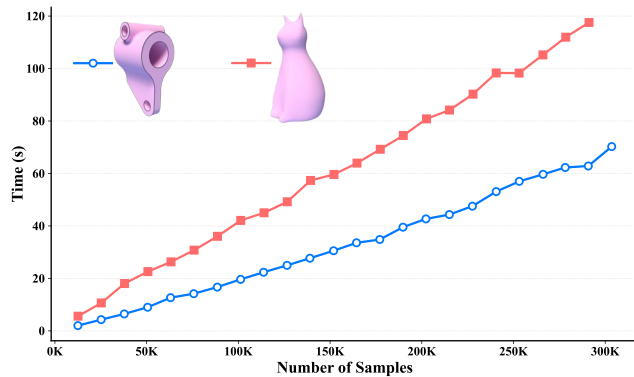


Fig. 11. Runtime scaling with increasing sample count (fixed simplification target of 1,000 vertices). The timing varies slightly across different models depending on their geometric complexity.

5.3 Performance Analysis

Our algorithm consists of several key stages: blue noise sampling, RVD computation, 3D Voronoi computation, priority queue initialization, and iterative edge collapse. Benefiting from the additivity property of SQEM matrices, our method achieves efficient computation times.

Table 1 presents the runtime breakdown for each stage on the models from Figures 8 and 10. The results show that edge simplification dominates the overall computation time, accounting for the largest proportion of the total runtime. This is expected, as the iterative collapse process involves repeated energy evaluations and priority queue updates.

Figure 11 illustrates how computation time scales with the number of input samples (with the medial axis consistently simplified to 1K vertices). The timing varies slightly across different models depending on their geometric complexity. For typical models, 50K sample points provide sufficient coverage, resulting in total computation times under 15 seconds. This demonstrates the practical efficiency of our approach—the surface-guided simplification avoids the expensive iterative reconstruction while maintaining high-quality results.

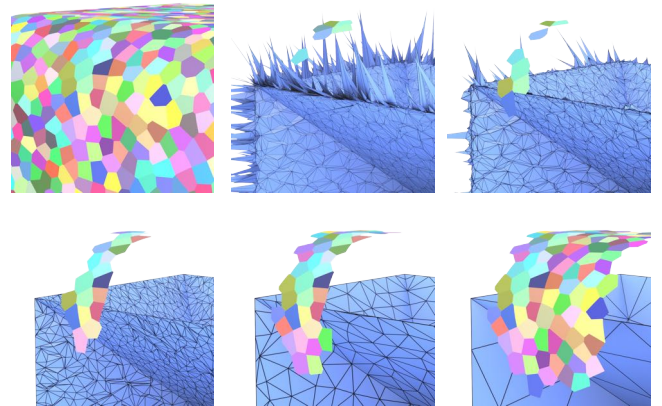


Fig. 12. Visualization of Atlas evolution across successive simplification stages on a filleted cube model. The top-left subfigure shows the local Restricted Voronoi Diagram (RVD) on the surface. The remaining five subfigures illustrate the surface regions associated with a medial mesh edge at different simplification stages, demonstrating how the Atlas progressively aggregates surface patches as edge collapses proceed.

5.4 Progressive Simplification

Our method begins with surface sampling and computes the 3D Voronoi diagram as the initial medial axis. Guided by the RVD correspondence, the algorithm progressively simplifies the medial structure through iterative edge collapses until reaching the target number of vertices. Figure 12 illustrates how the Atlas evolves across simplification stages on a filleted cube model. As simplification proceeds, the medial structure progressively converges toward the centerline of the fillet region, with the associated Atlas accurately capturing the corresponding surface patches.

Figure 13 illustrates this process on both a CAD model and an organic model, showing intermediate results from the initial medial axis to the final simplified structure, along with the bidirectional Hausdorff distance (HD) and sphere count at each stage. The initial medial axis contains numerous unwanted spike structures, exhibits poor triangle quality, and fails to accurately capture geometric

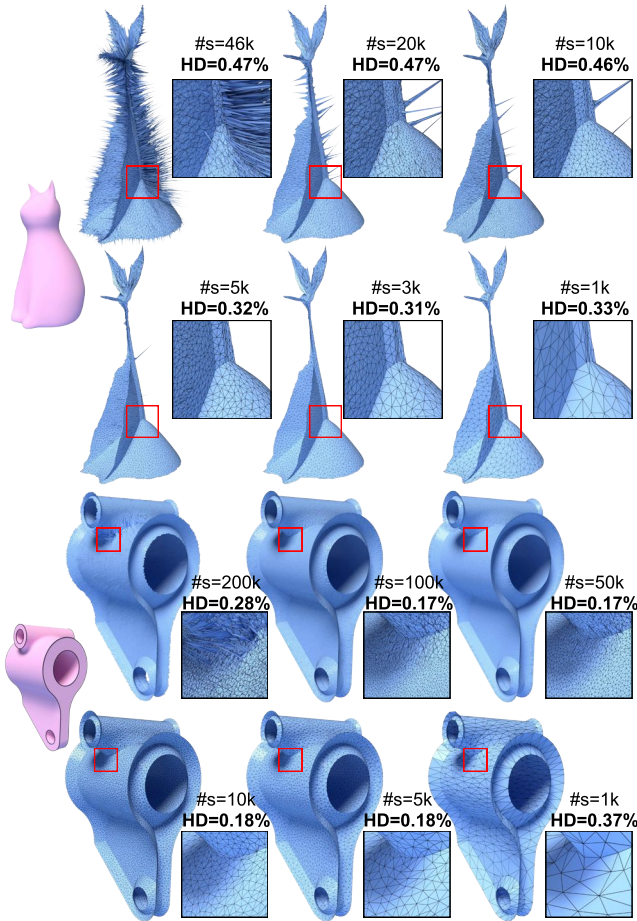


Fig. 13. Progressive simplification on a CAD model and an organic model. The initial medial axis contains numerous spurious spikes. As simplification progresses, these spikes are removed and triangle quality improves. Notably, the Hausdorff distance (HD) initially decreases before eventually increasing, demonstrating that our surface-guided simplification actively corrects the initial Voronoi structure—intermediate results achieve better approximation than the initialization. In the late stages, as the sphere count becomes very small, approximation capacity naturally declines.

features. As simplification progresses, these spurious spikes gradually disappear, triangle quality improves, and feature alignment becomes increasingly evident. Notably, the HD metric initially decreases before eventually increasing again, revealing a fundamental characteristic of our approach: rather than progressively refining an inherited medial structure, our method simplifies directly with respect to the original surface geometry. The intermediate simplified results actually achieve better surface approximation than the initial Voronoi-based medial axis, demonstrating that our surface-guided framework actively improves geometric fidelity during simplification. Only in the late stages, when the sphere count becomes very small, does approximation capacity naturally decline.

Figure 14 further demonstrates the approximation capability of our method when simplified to very few vertices on both CAD and

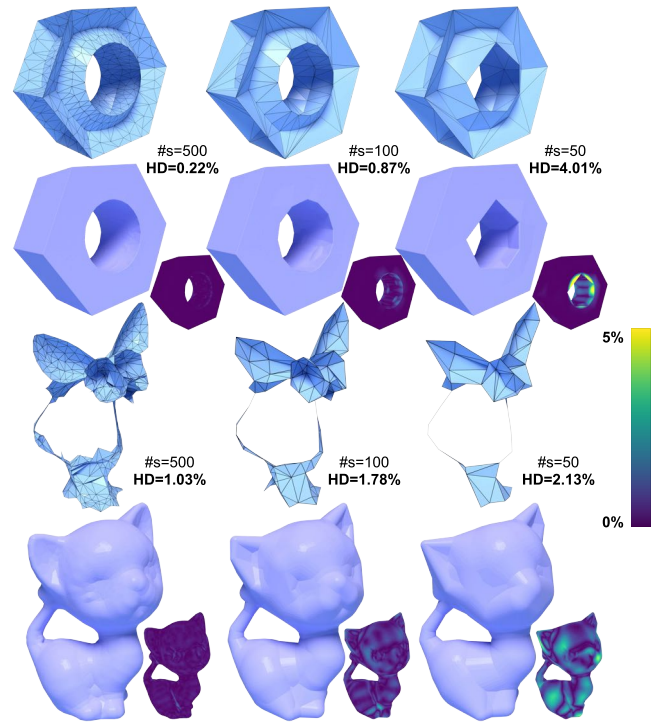


Fig. 14. Medial axis results simplified to low vertex counts. Despite the sparse sphere count, our method maintains meaningful structural correspondence and produces structurally valid representations even at aggressive simplification levels, demonstrating the effectiveness of the surface-guided optimization strategy.

organic models. Even with aggressive simplification, the medial axis retains meaningful geometric representation. To more clearly illustrate the reconstruction fidelity, we additionally provide a per-point Hausdorff distance visualization using a color-coded scalar field, where the color at each surface point reflects its distance to the reconstructed surface. The visualization confirms that reconstruction error remains well-distributed across the surface, with larger deviations concentrated in geometrically complex regions as the sphere count decreases.

5.5 Robustness to Noise

We further evaluate the robustness of our method on noisy inputs. Following standard practice, we add random displacements to each vertex of the input models (one CAD model and one organic model). The displacement magnitude for each vertex is randomly sampled from $[0, \eta \cdot d]$, where d is the diagonal length of the bounding box and η controls the maximum noise level. Figure 15 shows our results under different noise levels: no noise, $\eta = 0.25\%$, and $\eta = 0.5\%$.

When models contain noise, surface normals become unreliable—the local tangent planes defined by sample points and their normals no longer accurately represent the underlying geometry, rendering the plane-based SQEM energy inaccurate. However, our method demonstrates noise resilience. Noise introduces local normal discontinuities that naturally trigger our *invaginated cell* detection,

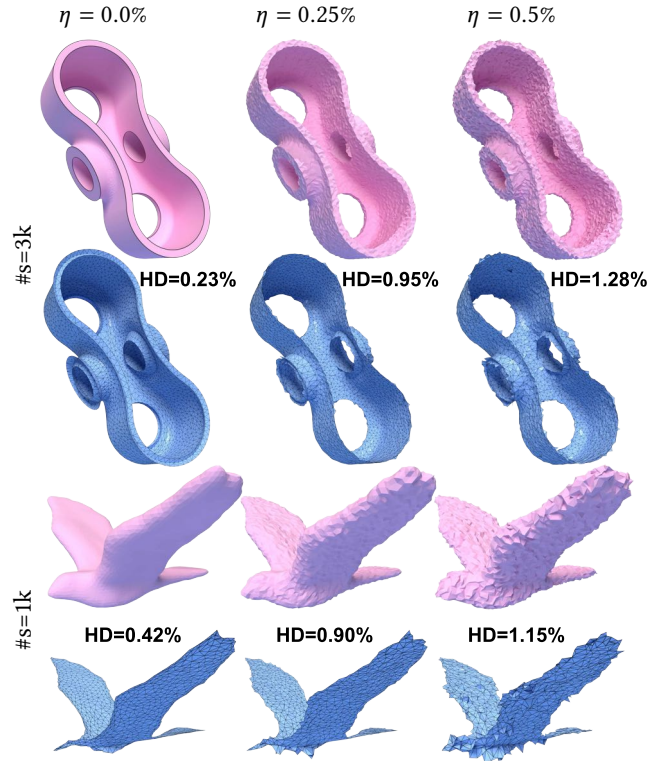


Fig. 15. Robustness to noise. We add random displacements with different magnitudes ($\eta = 0, 0.25\%, 0.5\%$) to test models. Noise introduces normal discontinuities that activate our Fidelity term for invaginated RVD cells, enabling our method to produce reasonable medial structures across the tested noise levels.

activating the Fidelity term for invaginated RVD cells. Since this formulation directly measures distances to sample sites rather than relying on tangent plane approximations, it provides more robust optimization under noisy conditions. As shown in Figure 15, our method produces reasonable medial structures across the tested noise levels.

5.6 Ablation Studies and Parameter Analysis

Component Ablation. We evaluate the contribution of key components in our method through ablation studies. Figure 16 shows results on a CAD model under four configurations: the full method, without Medial Face Filtering near Concave Features, without sharp feature preservation, and without Fidelity term for invaginated RVD cells.

Without Medial Face Filtering near Concave Features results in spurious complex structures in concave regions, as edge collapse alone cannot effectively remove non-medial faces in these areas. Without sharp feature preservation, external boundaries progressively collapse inward, failing to preserve the sharp feature information characteristic of CAD models. Disabling Fidelity term for invaginated RVD cells causes spheres to cluster excessively and

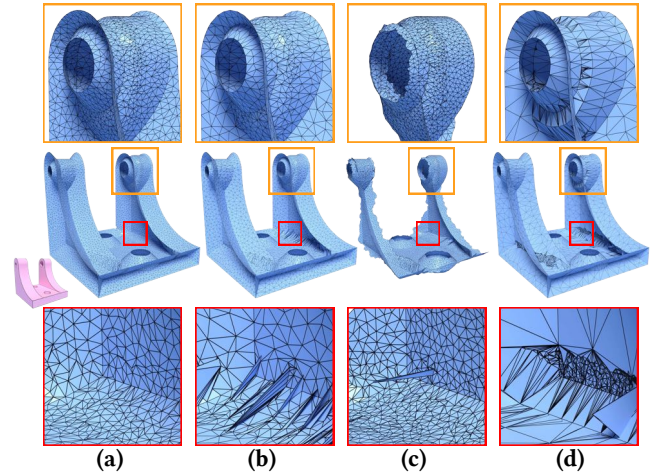


Fig. 16. Ablation study on a CAD model. (a) Full method. (b) Without Medial Face Filtering near Concave Features. (c) Without sharp feature preservation—external boundaries collapse inward, losing sharp feature alignment. (d) Without Fidelity term for invaginated RVD cells—spheres cluster excessively and produce incorrect medial structures.

produces incorrect medial structures in these areas. These ablations confirm that each component plays a critical role in achieving accurate results.

Parameter Sensitivity. We also examine the effect of two key parameters. The weight λ balances surface fidelity against mesh quality. Figure 17 shows results as λ varies from 10^{-7} to 1. Larger λ values prioritize mesh quality but reduce approximation fidelity, failing to accurately capture internal and external features. As λ decreases, features become increasingly prominent. When λ is too small, the method allocates excessive triangles to geometrically complex regions, degrading mesh quality.

The threshold τ determines the strictness of spike classification. Figure 18 shows results across different τ values. Without spike filtering, standard simplification criteria fail to remove spurious branches effectively. As τ increases, spike removal becomes more aggressive. However, when τ is too large, the method removes legitimate medial structures, weakening approximation capability.

Despite these variations, our method produces acceptable results across a reasonable parameter range, demonstrating robustness to parameter choices.

Fixed Parameters. Beyond λ and τ , our method involves several additional parameters that are fixed throughout all experiments. The dihedral angle tolerance ϕ controls the detection of sharp and concave features: we set $\phi = 45^\circ$ in all experiments, which provides reliable feature detection across a wide range of CAD models without requiring per-model tuning. The filtering threshold $\alpha = 0.7$ in the medial face filtering step (Equation 4) is also kept constant; values in the range $[0.6, 0.8]$ yield consistent results, and the method is not sensitive to moderate variations in this parameter. The topology preservation threshold is set to 200 vertices, below which the Link Condition is enforced prior to every edge collapse; this value is chosen to reduce the risk of undesired topological changes during

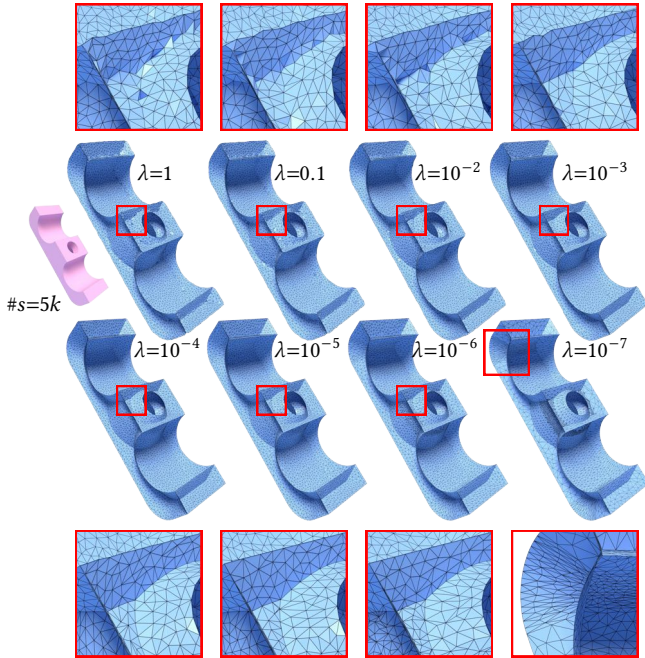


Fig. 17. Effect of the weight parameter λ on medial axis results. Larger λ values prioritize mesh quality but weaken feature capture—internal and external features become less distinct. As λ decreases, feature alignment becomes increasingly prominent. When λ is too small, mesh quality degrades as the method allocates excessive triangles to geometrically complex regions.

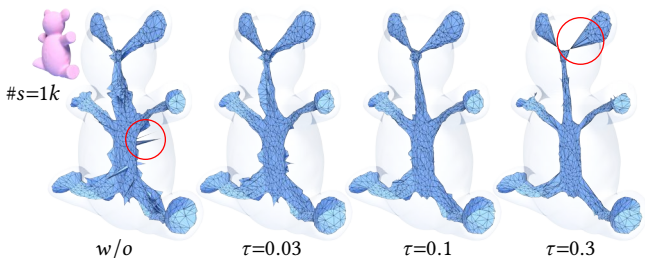


Fig. 18. Effect of the stability threshold parameter τ on spike removal. The parameter τ controls the strictness of spike classification. Without spike filtering, spurious branches cannot be reliably removed. As τ increases, spike removal becomes more aggressive. However, when τ is too large, legitimate medial structures are incorrectly removed, weakening approximation capability.

simplification, and can be increased for models with more complex topology. The sharpness parameter $k = 100$ in the sigmoid weighting function Ψ enforces a near-binary transition between spike removal and geometry-guided simplification; in practice, any sufficiently large value (e.g., $k \geq 50$) produces equivalent behavior. The number of surface samples is set to $10K$ for organic models and $50K$ for CAD models, which we find to provide sufficient geometric coverage for typical inputs.

Despite its general effectiveness, our method does encounter difficulties in certain challenging configurations. Figure 19 shows a representative failure case on a thin sheet-like model: insufficient sampling density on such thin surfaces causes the 3D Voronoi diagram to penetrate through the thin surface during initialization, resulting in holes in the medial structure, and also fails to capture side features accurately.

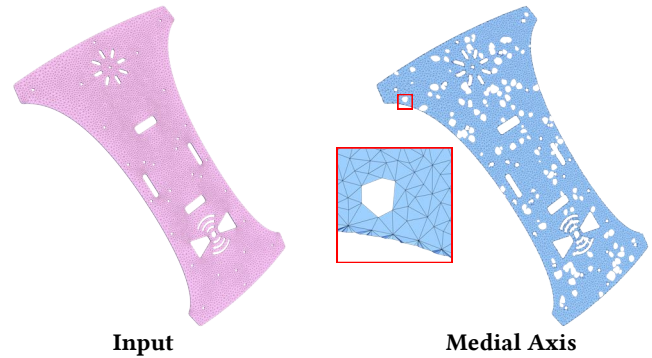


Fig. 19. A representative failure case on a thin sheet-like model. With $15K$ surface samples, the insufficient sampling density causes the 3D Voronoi diagram to penetrate through the thin surface during initialization, resulting in holes in the medial structure, and also prevents accurate capture of the side features.

5.7 Potential Application

Fillet Detection. Filletting is a fundamental operation in CAD systems, creating smooth transitions between adjacent surface patches—akin to a ball rolling between two surfaces to form a seamless connection. While fillets appear as narrow transitional regions on the surface, they play a crucial role in reverse engineering and secondary design phases. However, fillet detection remains challenging, particularly when input data originates from surface reconstruction or discretization processes [Jiang et al. 2025].

The medial axis provides a natural geometric lens for understanding fillets. While medial axes generally consist of both variable-radius and constant-radius branches, the medial structure of a fillet converges to the centerline traced by the rolling sphere centers—a well-defined curve with smoothly varying radius. Benefiting from our method’s accurate geometric capture capability, the computed medial axis precisely identifies these fillet centerlines. Figure 20 shows our results on two models containing variable-radius fillets, where the red-highlighted edges accurately trace the fillet centerlines. Our medial representation naturally encodes the information needed for fillet detection: each medial edge carries both radius values (from optimization) and explicit correspondence to surface regions (through the dual Voronoi framework). This enables a straightforward detection strategy—starting from medial boundary edges, we verify whether the Euclidean distances from corresponding surface triangles to the medial edge consistently match the optimized radius values. Edges satisfying this criterion are classified as fillet centerlines, as demonstrated in Figure 20. These results highlight the potential of our surface-guided framework for this important CAD analysis task.

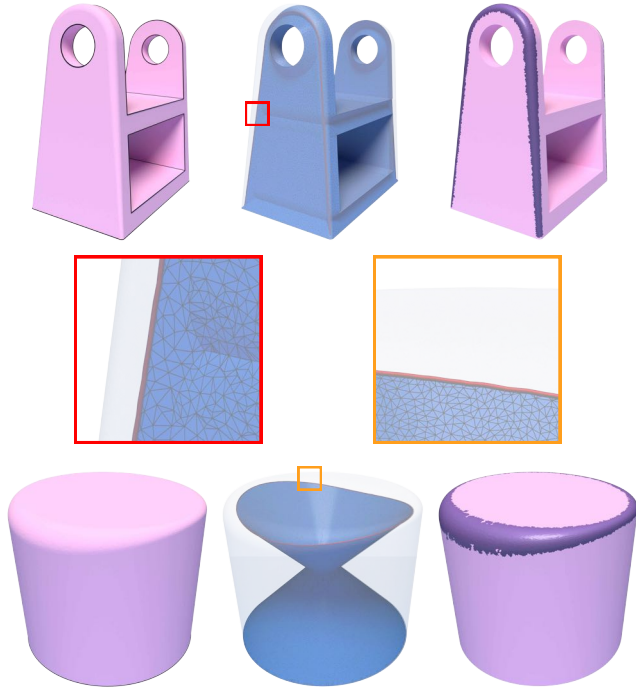


Fig. 20. Fillet detection application. Our surface-guided framework accurately captures fillet centerlines (red), including variable-radius cases. The natural encoding of radius values and surface correspondence in medial edges enables straightforward fillet detection through distance consistency verification.

Surface Extraction from Unsigned Distance Fields. Unsigned Distance Fields (UDFs) are a widely adopted shape representation in computer graphics and vision, supporting a broad range of applications including neural implicit modeling and shape reconstruction. A fundamental challenge in UDF-based pipelines is surface extraction: unlike signed distance fields, UDFs lack sign information and therefore cannot be directly processed by standard approaches such as Marching Cubes. One prominent line of work [Chen et al. 2025; Hou et al. 2023] addresses this by first extracting an ϵ -isosurface, yielding a double-layered structure that is subsequently projected toward the zero level set via optimization, followed by a post-processing step to recover a single-layer surface. However, the optimization is sensitive to parameter choices and prone to geometric artifacts in high-curvature regions, while the double-to-single-layer transition remains fragile and can fail on non-manifold or non-orientable inputs.

In this context, the medial axis of the double-layered ϵ -isosurface naturally corresponds to the underlying surface encoded by the UDF. This medial representation inherently supports non-manifold structures and directly yields a single-layer output, bypassing the need for explicit collapse optimization. We believe that a high-quality medial axis computation can serve as a principled and robust foundation for this extraction step.

To demonstrate this potential, we take a garment model from the Deep Fashion3D V2 dataset [Heming et al. 2020] as a representative example. We extract the ϵ -isosurface near the surface and apply several representative medial axis methods for comparison. As shown in Figure 21, our method produces a cleaner result with more regular boundaries and closer alignment to the ground truth surface, highlighting the promise of our surface-guided framework for UDF-based surface extraction.

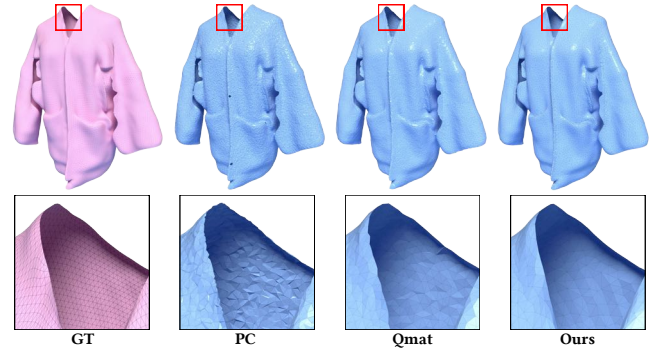


Fig. 21. Surface extraction from an unsigned distance field on a garment model from the Deep Fashion3D V2 dataset [Heming et al. 2020], comparing the ground truth surface against extraction results using three representative medial axis methods. Our result exhibits cleaner boundaries and closer geometric alignment to the ground truth, demonstrating the potential of our surface-guided framework for UDF-based surface extraction.

6 Limitations and Future Work

Our method has several limitations that suggest directions for future research. First, the 3D Voronoi-based initialization faces challenges with extremely thin sheet-like models, where insufficient thickness may prevent the formation of a valid initial medial structure. Additionally, the simplified medial mesh may exhibit slight self-intersections, particularly in regions with complex branching or high curvature. Finally, our current topology preservation strategy is heuristic in nature and does not provide formal guarantees; topological errors may still occur, particularly on models with high topological complexity.

Looking forward, we plan to investigate more robust initialization strategies that can handle extremely thin geometric features, potentially through adaptive sampling or alternative initialization schemes. We also intend to explore more principled topology preservation mechanisms that can provide stronger guarantees during simplification. Addressing these challenges would further enhance the method's applicability to a broader range of geometric modeling and analysis tasks.

7 Conclusion

We have presented Structural MAT, a robust framework for medial axis simplification that maintains explicit correspondence between medial axis vertices and surface regions throughout the entire simplification process. By tracking the association between each 3D medial element and its corresponding surface patch—what we term

the Atlas—our method achieves clean boundaries, accurate feature preservation, and high computational scalability. Our approach naturally handles challenging geometric features including sharp edges and filleted surfaces, which have traditionally been difficult for medial axis methods, making it particularly effective for both organic shapes and precision-engineered CAD models.

Extensive evaluation demonstrates that Structural MAT consistently outperforms existing methods across diverse geometric datasets, achieving superior shape fidelity while maintaining computational efficiency. The demonstrated application in fillet detection showcases the utility of our surface-aware representation for high-level geometric reasoning tasks, confirming that explicit surface correspondence serves as a powerful foundation for downstream applications in computer-aided design and manufacturing.

Acknowledgments

The authors thank the anonymous reviewers for their insightful comments and suggestions. This work was supported by the National Natural Science Foundation of China (Key Project Grants 12494550 and 12494553; Grants U23A20312 and 62272277) and the Natural Science Foundation of Shandong Province (Grant ZR2025MS986).

References

- Nina Amenta, Sunghye Choi, and Ravi Krishna Kolluri. 2001. The power crust, unions of balls, and the medial axis transform. *Computational Geometry* 19, 2-3 (2001), 127–153.
- Dominique Attali and Annick Montanvert. 1996. Modeling noise for a better simplification of skeletons. In *Proceedings of 3rd IEEE International Conference on Image Processing*, Vol. 3. IEEE, 13–16.
- Jonathan W Brandt. 1994. Convergence and continuity criteria for discrete approximations of the continuous planar skeleton. *CVGIP: Image Understanding* 59, 1 (1994), 116–124.
- Frédéric Chazal and André Lieutier. 2005. The “ λ -medial axis”. *Graphical models* 67, 4 (2005), 304–331.
- Frédéric Chazal and André Lieutier. 2008. Smooth manifold reconstruction from noisy and non-uniform approximation with guarantees. *Computational Geometry* 40, 2 (2008), 156–170.
- Xuhui Chen, Fugang Yu, Fei Hou, Wencheng Wang, Zhebin Zhang, and Ying He. 2025. DCUDF2: Improving Efficiency and Accuracy in Extracting Zero Level Sets From Unsigned Distance Fields. *IEEE Transactions on Visualization and Computer Graphics* 31, 10 (2025), 9052–9065. doi:10.1109/TVCG.2025.3588659
- Tamal K. Dey, Herbert Edelsbrunner, Sumanta Guha, and Dmitry V. Nekhayev. 1998. Topology preserving edge contraction. <https://api.semanticscholar.org/CorpusID:50333417>
- Tamal K Dey and Wulue Zhao. 2004. Approximating the medial axis from the Voronoi diagram with a convergence guarantee. *Algorithmica* 38, 1 (2004), 179–200.
- Zhiyang Dou, Cheng Lin, Rui Xu, Lei Yang, Shiqing Xin, Taku Komura, and Wenping Wang. 2022. Coverage axis: Inner point selection for 3d shape skeletonization. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 419–432.
- Zhiyang Dou, Qingxuan Wu, Cheng Lin, Zeyu Cao, Qiangqiang Wu, Weilin Wan, Taku Komura, and Wenping Wang. 2023. Tore: Token reduction for efficient human mesh recovery with transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15143–15155.
- Zhiyang Dou, Shiqing Xin, Rui Xu, Jian Xu, Yuanfeng Zhou, Shuangmin Chen, Wenping Wang, Xiuyang Zhao, and Changhe Tu. 2020. Top-down shape abstraction based on greedy pole selection. *IEEE transactions on visualization and computer graphics* 27, 10 (2020), 3982–3993.
- Zhiyang Fu, Rui Xu, Shiqing Xin, Shuangmin Chen, Changhe Tu, Chenglei Yang, and Lin Lu. 2022. Easyvrmodeling: Easily create 3d models by an immersive vr system. *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 5, 1 (2022), 1–14.
- Michael Garland and Paul S. Heckbert. 1997. Surface simplification using quadric error metrics. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '97)*. ACM Press/Addison-Wesley Publishing Co., USA, 209–216. doi:10.1145/258734.258849
- Joachim Giesen, Balint Miklos, Mark Pauly, and Camille Wormser. 2009. The scale axis transform. In *Proceedings of the twenty-fifth annual symposium on Computational geometry*. 106–115.
- Zhu Heming, Cao Yu, Jin Hang, Chen Weikai, Du Dong, Wang Zhangye, Cui Shuguang, and Han Xiaoguang. 2020. Deep Fashion3D: A Dataset and Benchmark for 3D Garment Reconstruction from Single Images. In *Computer Vision – ECCV 2020*. Springer International Publishing, 512–530.
- Wim H Hesselink and Jos BTM Roerdink. 2008. Euclidean skeletons of digital image and volume data in linear time by the integer medial axis transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 12 (2008), 2204–2217.
- Fei Hou, Xuhui Chen, Wencheng Wang, Hong Qin, and Ying He. 2023. Robust Zero Level-Set Extraction from Unsigned Distance Fields Based on Double Covering. *ACM Trans. Graph.* 42, 6, Article 245 (dec 2023), 15 pages. doi:10.1145/3618314
- Jianwei Hu, Bin Wang, Lihui Qian, Yiling Pan, Xiaohu Guo, Lingjie Liu, and Wenping Wang. 2019. MAT-Net: Medial Axis Transform Network for 3D Object Recognition.. In *IJCAI*. 774–781.
- Qijia Huang, Pierre Kraemer, Sylvain Thery, and Dominique Bechmann. 2024. Dynamic Skeletonization via Variational Medial Axis Sampling. In *SIGGRAPH Asia 2024 Conference Papers* (Tokyo, Japan) (SA '24). Association for Computing Machinery, New York, NY, USA, Article 66, 11 pages. doi:10.1145/3680528.3687678
- Clément Jamin, Sylvain Pion, and Monique Teillaud. 2025. 3D Triangulations. In *CGAL User and Reference Manual* (6.1 ed.). CGAL Editorial Board. <https://doc.cgal.org/6.1/Manual/packages.html#PkgTriangulation3>
- Jing-En Jiang, Hanxiao Wang, Mingyang Zhao, Dong-Ming Yan, Shuangmin Chen, Shiqing Xin, Changhe Tu, and Wenping Wang. 2025. DeFillet: Detection and Removal of Fillet Regions in Polygonal CAD Models. *ACM Trans. Graph.* 44, 4, Article 115 (July 2025), 19 pages. doi:10.1145/3731166
- Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. 2019. ABC: A Big CAD Model Dataset For Geometric Deep Learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Lei Lan, Ran Luo, Marco Fratarcangeli, Weiwei Xu, Huamin Wang, Xiaohu Guo, Junfeng Yao, and Yin Yang. 2020. Medial elastics: Efficient and collision-ready deformation via medial axis transform. *ACM Transactions on Graphics (TOG)* 39, 3 (2020), 1–17.
- Lei Lan, Yin Yang, Danny Kaufman, Junfeng Yao, Minchen Li, and Chenfanju Jiang. 2021. Medial IPC: accelerated incremental potential contact with medial elastics. *ACM Transactions on Graphics* 40, 4 (2021).
- Bruno Lévy. 2026. *GraphiteThree: An Experimental 3D Modeler*. <https://github.com/BrunoLevy/GraphiteThree> Built around the Geogram library.
- Bruno Lévy and Nicolas Bonneel. 2013. Variational Anisotropic Surface Meshing with Voronoi Parallel Linear Enumeration. In *Proceedings of the 21st International Meshing Roundtable*, Xiangmin Jiao and Jean-Christophe Weill (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 349–366.
- Pan Li, Bin Wang, Feng Sun, Xiaohu Guo, Caiming Zhang, and Wenping Wang. 2016. Q-MAT: Computing Medial Axis Transform By Quadratic Error Minimization. *ACM Trans. Graph.* 35, 1, Article 8 (Dec. 2016), 16 pages. doi:10.1145/2753755
- Cheng Lin, Changjian Li, Yuan Liu, Nenglung Chen, Yi-King Choi, and Wenping Wang. 2021. Point2skeleton: Learning skeletal representations from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4277–4286.
- Cheng Lin, Lingjie Liu, Changjian Li, Leif Kobbelt, Bin Wang, Shiqing Xin, and Wenping Wang. 2020. Seg-mat: 3d shape segmentation using medial axis transform. *IEEE transactions on visualization and computer graphics* 28, 6 (2020), 2430–2444.
- Dong C. Liu and Jorge Nocedal. 1989. On the limited memory BFGS method for large scale optimization. *Mathematical Programming* 45 (1989), 503–528. <https://api.semanticscholar.org/CorpusID:5681609>
- Balint Miklos, Joachim Giesen, and Mark Pauly. 2010. Discrete scale axis representations for 3D geometry. In *ACM SIGGRAPH 2010 papers*. 1–10.
- Yuta Noma, Silvia Sellán, Nicholas Sharp, Karan Singh, and Alec Jacobson. 2024. Surface-filling curve flows via implicit medial axes. *ACM Transactions on Graphics (TOG)* 43, 4 (2024), 1–12.
- Dmitry Petrov, Pradyumn Goyal, Vikas Thamizharasan, Vladimir Kim, Matheus Gadelha, Melinos Averkiou, Siddhartha Chaudhuri, and Evangelos Kalogerakis. 2024. Gem3d: Generative medial abstractions for 3d shape synthesis. In *ACM SIGGRAPH 2024 Conference Papers*. 1–11.
- Chris Pudney. 1998. Distance-ordered homotopic thinning: a skeletonization algorithm for 3D digital images. *Computer vision and image understanding* 72, 3 (1998), 404–413.
- Martin Rumpf and Alexandru Telea. 2002. A continuous skeletonization method based on level sets. In *EPRINTS-BOOK-TITLE*. University of Groningen, Johann Bernoulli Institute for Mathematics and ...
- Rohan Sawhney. 2021. *FCPW: Fastest Closest Points in the West*.
- Kaleem Siddiqi and Stephen Pizer. 2008. *Medial representations: mathematics, algorithms and applications*. Vol. 37. Springer Science & Business Media.
- Shibo Song and Ningna Wang. 2023. blender-mat-addon. <https://github.com/songshibo/blender-mat-addon>.
- Andrea Tagliasacchi, Thomas Delame, Michela Spagnuolo, Nina Amenta, and Alexandru Telea. 2016. 3d skeletons: A state-of-the-art report. In *Computer Graphics Forum*, Vol. 35. Wiley Online Library, 573–597.

- Jean-Marc Thiery, Émilie Guy, and Tamy Boubekeur. 2013. Sphere-meshes: Shape approximation using spherical quadric error metrics. *ACM Transactions on Graphics (TOG)* 32, 6 (2013), 1–12.
- Ningna Wang, Hui Huang, Shibo Song, Bin Wang, Wenping Wang, and Xiaohu Guo. 2024b. MATTopo: Topology-preserving Medial Axis Transform with Restricted Power Diagram. *ACM Transactions on Graphics (TOG)* 43, 4 (2024). doi:10.1145/3687763
- Ningna Wang, Bin Wang, Wenping Wang, and Xiaohu Guo. 2022. Computing medial axis transform with feature preservation via restricted power diagram. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)* 41, 6 (2022), 1–18.
- Ningna Wang, Rui Xu, Yibo Yin, Zichun Zhong, Taku Komura, Wenping Wang, and Xiaohu Guo. 2025c. MATStruct: High-quality Medial Mesh Computation via Structure-aware Variational Optimization. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers (SA Conference Papers '25)*. Association for Computing Machinery, New York, NY, USA, Article 170, 12 pages. doi:10.1145/3757377.3763840
- Pengfei Wang, Jiantao Song, Lei Wang, Shiqing Xin, Dong-Ming Yan, Shuangmin Chen, Changhe Tu, and Wenping Wang. 2025a. Towards Voronoi Diagrams of Surface Patches. *IEEE Transactions on Visualization and Computer Graphics* 31, 10 (2025), 6810–6823. doi:10.1109/TVCG.2025.3531445
- Pengfei Wang, Jiantao Song, Shiqing Xin, Shuangmin Chen, Changhe Tu, Wenping Wang, and Jiaye Wang. 2025b. Efficient Nearest Neighbor Search Using Dynamic Programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2025), 1–16. doi:10.1109/TPAMI.2025.3610211
- Zimeng Wang, Zhiyang Dou, Rui Xu, Cheng Lin, Yuan Liu, Xiaoxiao Long, Shiqing Xin, Taku Komura, Xiaoming Yuan, and Wenping Wang. 2024a. Coverage axis++: Efficient inner point selection for 3D shape skeletonization. In *Computer Graphics Forum*, Vol. 43. Wiley Online Library, e15143.
- Shiqing Xin, Pengfei Wang, Rui Xu, Dongming Yan, Shuangmin Chen, Wenping Wang, Caiming Zhang, and Changhe Tu. 2022. SurfaceVoronoi: Efficiently Computing Voronoi Diagrams Over Mesh Surfaces with Arbitrary Distance Solvers. *ACM Trans. Graph.* 41, 6, Article 185 (Nov. 2022), 12 pages. doi:10.1145/3550454.3555453
- Dong-Ming Yan, Bruno Lévy, Yang Liu, Feng Sun, and Wenping Wang. 2009. Isotropic Remeshing with Fast and Exact Computation of Restricted Voronoi Diagram. *Computer Graphics Forum* 28, 5 (2009), 1445–1454. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8659.2009.01521.x doi:10.1111/j.1467-8659.2009.01521.x
- Yajie Yan, David Letscher, and Tao Ju. 2018. Voxel cores: Efficient, robust, and provably good approximation of 3d medial axes. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–13.
- Yajie Yan, Kyle Sykes, Erin Chambers, David Letscher, and Tao Ju. 2016. Erosion thickness on medial axes of 3D shapes. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–12.
- Baorong Yang, Junfeng Yao, Bin Wang, Jianwei Hu, Yiling Pan, Tianxiang Pan, Wenping Wang, and Xiaohu Guo. 2020. P2MAT-NET: Learning medial axis transform from sparse point clouds. *Computer Aided Geometric Design* 80 (2020), 101874.
- Yiding Yang, Zhou Ren, Haoxiang Li, Chunluan Zhou, Xinchao Wang, and Gang Hua. 2021. Learning dynamics via graph neural networks for human pose estimation and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8074–8084.
- Qingnan Zhou and Alec Jacobson. 2016. Thing10K: A Dataset of 10,000 3D-Printing Models. *arXiv preprint arXiv:1605.04797* (2016).
- Yang Zhou, Kangxue Yin, Hui Huang, Hao Zhang, Minglun Gong, and Daniel Cohen-Or. 2015. Generalized cylinder decomposition. *ACM Trans. Graph.* 34, 6 (2015), 171–1.

A Gradient Derivation and Closed-Form Solution

This appendix provides the explicit gradients required by the L-BFGS optimization in Section 4.2, as well as the closed-form solution available when no invaginated cells are involved. L-BFGS requires only first-order gradient information, so no explicit Hessian is derived.

Optimization variables. The optimization is performed over $\bar{v} = (v_x, v_y, v_z, r)^\top \in \mathbb{R}^4$, i.e., the sphere center v and radius r . All surface-side quantities (s_i , p , \mathbf{n} , and the associated region \mathcal{A}) are fixed constants inherited at the moment of edge collapse.

Facet-level decomposition. The surface region within each RVD cell is composed of planar triangle facets. By linearity of differentiation, the total gradient is the sum of per-facet gradients:

$$\nabla_{\bar{v}} E_{\text{Fidelity}}(\bar{v}) = \sum_{T \in \mathcal{A}} \nabla_{\bar{v}} E_T(\bar{v}), \quad (19)$$

where T denotes a triangle facet with area $|T|$. We treat regular and invaginated cells separately below.

A.1 Regular Cells: Quadratic Form and Gradient

For a facet T in a regular cell with unit normal \mathbf{n} , the per-facet energy reduces to a quadratic form in \bar{v} :

$$E_T(\bar{v}) = \bar{v}^\top \mathbf{A}_T \bar{v} + \mathbf{b}_T^\top \bar{v} + c_T, \quad (20)$$

with coefficients

$$\mathbf{A}_T = |T| \begin{pmatrix} \mathbf{nn}^\top & \mathbf{n} \\ \mathbf{n}^\top & 1 \end{pmatrix}, \quad \mathbf{b}_T = -2|T|(\mathbf{n}^\top \bar{p}_T) \begin{pmatrix} \mathbf{n} \\ 1 \end{pmatrix}, \quad c_T = \int_T (\mathbf{n}^\top p)^2 dA, \quad (21)$$

where \bar{p}_T is the centroid of T . The gradient is linear:

$$\nabla_{\bar{v}} E_T(\bar{v}) = 2\mathbf{A}_T \bar{v} + \mathbf{b}_T. \quad (22)$$

By additivity, contributions from all regular facets can be pre-aggregated once into global coefficients before optimization:

$$\mathbf{A}_{\text{reg}} = \sum_{T \in \mathcal{A}_{\text{reg}}} \mathbf{A}_T, \quad \mathbf{b}_{\text{reg}} = \sum_{T \in \mathcal{A}_{\text{reg}}} \mathbf{b}_T. \quad (23)$$

A.2 Invaginated Cells: Nonlinear Gradient

For a facet T in an invaginated cell with generating site s_i , the energy $E_T(\bar{v}) = |T|(\|v - s_i\| - r)^2$ is nonlinear. Computing its gradient:

$$\frac{\partial E_T}{\partial v} = 2|T|(\|v - s_i\| - r) \cdot \frac{v - s_i}{\|v - s_i\|}, \quad \frac{\partial E_T}{\partial r} = 2|T|(r - \|v - s_i\|). \quad (24)$$

Stacked as a 4D gradient:

$$\nabla_{\bar{v}} E_T(\bar{v}) = 2|T|(\|v - s_i\| - r) \begin{pmatrix} (v - s_i)/\|v - s_i\| \\ -1 \end{pmatrix}. \quad (25)$$

This gradient depends on the current \bar{v} through both the scalar factor and the direction, so it is recomputed at every L-BFGS iteration. For numerical stability, the denominator $\|v - s_i\|$ is clamped to $\max(\|v - s_i\|, \epsilon)$ with $\epsilon = 10^{-8}$.

A.3 Topological Laplacian Smoothing Gradient

The Laplacian term is quadratic in v and independent of r :

$$\frac{\partial E_{\text{Lap}}}{\partial v} = \frac{2}{|\mathcal{N}(v_1, v_2)|} \sum_{u \in \mathcal{N}(v_1, v_2)} (v - u), \quad \frac{\partial E_{\text{Lap}}}{\partial r} = 0. \quad (26)$$

A.4 Closed-Form Solution in the Purely Quadratic Case

When \mathcal{A} contains no invaginated cells, the total energy is purely quadratic in \bar{v} and can be solved in closed form without L-BFGS. Writing the Laplacian term in matrix form:

$$E_{\text{Lap}}(\bar{v}) = \bar{v}^\top \mathbf{A}_{\text{Lap}} \bar{v} + \mathbf{b}_{\text{Lap}}^\top \bar{v} + c_{\text{Lap}}, \quad (27)$$

with

$$\mathbf{A}_{\text{Lap}} = \begin{pmatrix} \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0}^\top & 0 \end{pmatrix}, \quad \mathbf{b}_{\text{Lap}} = -\frac{2}{|\mathcal{N}(v_1, v_2)|} \sum_{u \in \mathcal{N}(v_1, v_2)} \begin{pmatrix} u \\ 0 \end{pmatrix}, \quad (28)$$

the total energy becomes

$$E(\bar{v}) = \bar{v}^\top \mathbf{A}_{\text{tot}} \bar{v} + \mathbf{b}_{\text{tot}}^\top \bar{v} + \text{const}, \quad (29)$$

where

$$\mathbf{A}_{\text{tot}} = \mathbf{A}_{\text{reg}} + \lambda \mathbf{A}_{\text{Lap}}, \quad \mathbf{b}_{\text{tot}} = \mathbf{b}_{\text{reg}} + \lambda \mathbf{b}_{\text{Lap}}. \quad (30)$$

Setting $\nabla_{\bar{v}}E = 2\mathbf{A}_{\text{tot}}\bar{v} + \mathbf{b}_{\text{tot}} = \mathbf{0}$ yields the closed-form minimizer:

$$\bar{v}^* = -\frac{1}{2}\mathbf{A}_{\text{tot}}^{-1}\mathbf{b}_{\text{tot}}. \quad (31)$$

Since $\mathbf{A}_{\text{tot}} \in \mathbb{R}^{4 \times 4}$, Eq. 31 is obtained by solving a small 4×4 linear system and incurs negligible cost per edge collapse.

A.5 Total Gradient in the Nonlinear Case

When \mathcal{A} contains invaginated cells, the total gradient used by L-BFGS is assembled as:

$$\begin{aligned} \nabla_{\bar{v}}E(\bar{v}) = & \underbrace{2\mathbf{A}_{\text{reg}}\bar{v} + \mathbf{b}_{\text{reg}}}_{\text{regular (pre-aggregated)}} \\ & + \underbrace{\sum_{T \in \mathcal{A}_{\text{inv}}} 2|T|(\|v - s_{i(T)}\| - r) \begin{pmatrix} (v - s_{i(T)})/\|v - s_{i(T)}\| \\ -1 \end{pmatrix}}_{\text{invaginated (per-iteration)}} \\ & + \lambda \begin{pmatrix} \frac{2}{|\mathcal{N}(v_1, v_2)|} \sum_{u \in \mathcal{N}(v_1, v_2)} (v - u) \\ 0 \end{pmatrix}. \end{aligned} \quad (32)$$

where $s_{i(T)}$ is the generating site of the invaginated cell containing T . The per-iteration cost is proportional to $|\mathcal{A}_{\text{inv}}| + |\mathcal{N}(v_1, v_2)|$.