

Triangle Mesh Skeletonization Using Non-Deterministic Voxel Thinning and Graph Spectrum Segmentation.

Santiago Moreno-Avendano¹, Daniel Mejia-Parra^{1,2,*} and Oscar Ruiz-Salguero¹

¹CAD CAM CAE Laboratory, EAFIT University, Colombia

²Vicomtech Foundation, Basque Research and Technology Alliance (BRTA), Spain

*Corresponding author e-mail: dmejia@vicomtech.org

Abstract. In the context of shape processing, the estimation of the medial axis is relevant for the simplification and re-parameterization of 3D bodies. The currently used methods are based on (1) General fields, (2) Geometric methods and (3) voxel-based thinning. They present shortcomings such as (1) overrepresentation and non-smoothness of the medial axis due to high frequency nodes and (2) biased-skeletons due to skewed thinning. To partially overcome these limitations, this article presents a non-deterministic algorithm for the estimation of the 1D skeleton of triangular B-Reps or voxel-based body representations. Our method articulates (1) a novel randomized thinning algorithm that avoids possible skewings in the final skeletonization, (2) spectral-based segmentation that eliminates short dead-end branches, and (3) a maximal excursion method for reduction of high frequencies. The test results show that the randomized order in the removal of the instantaneous skin of the solid region eliminates bias of the skeleton, thus respecting features of the initial solid. An Alpha Shape-based inversion of the skeleton encoding results in triangular boundary Representations of the original body, which present reasonable quality for fast non-minute scenes. Future work is needed to (a) tune the spectral filtering of high frequencies off the basic skeleton and (b) extend the algorithm to solid regions whose skeletons mix 1D and 2D entities.

Keywords: Skeletonization, 3D thinning, 1D medial Axis, Randomization, Spectral segmentation, Graph representation.

1. Introduction

The medial axis (MA) of a compact 3D region $\Omega \subset R^3$, is defined as the set of all points $x \in \Omega$ such that the closest point in the boundary $\partial\Omega$ is not unique. Such a medial axis is unique, defined as the finite union of compact 1- and 2-manifold sets. An approximation of the $MA(\Omega)$ is called an skeleton $SK(\Omega)$. Some applications for skeletonization algorithms include shape decomposition, animation, medical images, virtual navigation and computational mechanics [18,20]. This manuscript presents a skeletonization algorithm based on non-deterministic voxel thinning and graph spectral decomposition which articulates: (1) a novel non-deterministic thinning algorithm for computing non-skewed 1-manifold $SK(\Omega)$, (2) spectral graph segmentation for clearing small death branches, and (3) application of a maximal excursion algorithm to smooth the resulting skeleton and reduce high frequencies. Finally, our algorithm applies Alpha-Shape inversion on the computed skeleton $SK(\Omega)$, producing a retrieved surface M_L that is an approximation of the input body surface $M = (X, T)$. The remainder of this article is organized as follows: Section 2 discusses the skeletonization methods

already present in the literature. Section 3 discusses our skeletonization method. Section 4 presents skeletonization and alpha-shape inversion results for different triangle meshes. Finally, Section 5 presents the conclusions and introduces what remains for future work.

2. Literature Review

Skeletonization methods are algorithms that produce an estimation of the Medial Axis of a 3D solid body that provides a lower dimensional (1D and 2D) representation of the original model. Depending on the approach for this dimensional reduction, skeletonization algorithms can be classified as: (1) general-field methods, (2) geometric methods and (3) voxel-thinning methods. The state of the art for the afore mentioned methodologies is briefly discussed below. For a more detailed survey, we refer the reader to [22].

General Field methods estimate the medial axis by computing a map field on Ω . The skeleton of Ω is extracted by connecting singularities on the map field [11,22]. Refs [2,23] compute a distance field map that have proven to be efficient and becoming suitable for large datasets [4]. Refs [10,22] compute potential fields that improves the robustness of the methods in the presence of input noisy shapes. Other methods use projection planes and inscribed balls to find the 2D curve skeleton. Afterwards, a convex hull operation is applied to generate the 1D skeleton [14]. General field methods can be applied to mesh and voxel data inputs [14,22]. In addition, they can produce also 2-manifold skeletons [2,22]. However, they (1) do not ensure the connectivity of the resulting skeleton [10,23] and, (2) they produce non-smooth skeletons with small dead-branches [19,20,22].

Geometric methods find the symmetries and geometric properties in the boundary of a shape region to generate the medial axis [19,20]. Voronoi diagrams [7], mesh contraction [25], octree-graphs [3], and union-of-balls [16] are popular geometric methods to compute the skeleton. These methods are applied to mesh or point-cloud representations [20], preserving the connectivity of the resulting skeleton [25] and generating non-skewed skeletons [25]. However, these methods still produce skeletons with small dead-branches and high frequencies [19,20,22].

Voxel-based thinning methods iteratively remove the boundary set of a voxel representation of the original region [6,11,18]. These algorithms usually preserve the connectivity of the skeleton by defining explicit rules such as: simple point criterion [15,17] or removing and recovering templates [12]. In addition, some thinning methods are able to produce 2D (surface) skeletons [11]. Current thinning-based methods present shortcomings such as: (1) computational expensive algorithms, (2) biased skeletons [13,27], and (3) non-smooth skeletons with small dead-branches [19,20,22]. Some authors have reduced the computational cost of these algorithms by applying parallel programming [6,12,27]. However, the resulting skeleton does not preserve the connectivity.

2.1. Conclusions of the literature review

Based on the literature review, the algorithms for skeletonization can be classified in: (1) General field methods are efficient for large datasets, robust to noisy shapes and some methods produce curve and surface skeletons but can present problems like discontinuities, (2) Geometric method conserve connectivity and generate well centered skeletons, and (3) Voxel-based thinning can be robust to noisy shapes, preserve connectivity and some methods produce curve and surface skeletons but are computationally expensive and the skeletons are skewed. Also, the surveys show that the most part of the algorithms are not smoothed and/or produce small death branches. To partially overcome this problem, this article presents (1) a random labelling that neutralizes the bias, (2) deletion of the small death branches using a spectral segmentation of the graph skeleton to improve the centricity and smoothness of the skeleton and (3) the use of maximal excursion algorithm to improve the smoothness. We also perform an alpha-shape-based inversion that retrieve an approximation of the original body.

3. Methodology

This section discusses the method used to compute the skeleton that approximates the Medial Axis $MA(\Omega)$ of a compact and connected 3D solid $\Omega \subset R^3$. Our method requires a triangle mesh $M = (X, T)$ as input. The mesh M discretizes the boundary $\partial\Omega$ of the solid body Ω . M is a closed 2-manifold defined by the set of points X and the set of triangles T . The proposed method proceeds as follows: (1) Our algorithm converts M into a voxel representation $Vox(\Omega)_0$, where a point $x \in \Omega$ is called a black voxel and a point $x \in R^3 \setminus \Omega$ is called white voxel. (2) A voxel-based thinning algorithm is applied to compute $SK(\Omega)$. In order to neutralize the skeleton bias of Voxel-based algorithms, a random labelling is applied to the voxelization in this step. The Graph Laplacian L of $SK(\Omega)$ and its corresponding spectral decomposition are computed. The spectral decomposition enables a segmentation of the skeleton in the frequency domain, which characterizes the large branches of $SK(\Omega)$. (4) Aided by the spectral segmentation, a shortest path algorithm is applied to $SK(\Omega)$, resulting in the suppression of the small death-branches. (5) A maximal excursion algorithm is applied to reduce high frequencies and improve the smoothness of $SK(\Omega)$. (6) Finally, an alpha-shape-based inversion is used to retrieve the original body Ω . Figure 1 shows the flowchart of the proposed algorithm. The remainder of this section describes the details of the skeletonization algorithm.

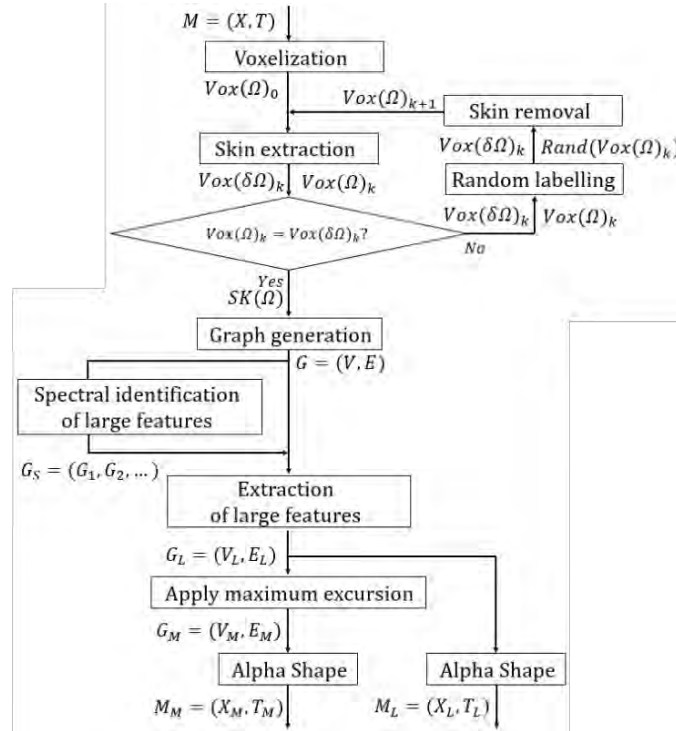


Figure 1. Flowchart of the proposed skeletonization method. Our algorithm articulates: (a) random labelling, (b) spectral segmentation, (c) skeleton smoothing and, (d) alpha-shape-based inversion.

3.1 Voxel-based thinning

In order to apply a thinning algorithm to the solid body Ω , the triangle mesh $M = (X, T)$ is converted into a voxel representation $Vox(\Omega)_0$. For this purpose, the voxelization algorithm presented in [1] is implemented. Thinning algorithms iteratively detect the boundary voxels $Vox(\partial\Omega)$ of $Vox(\Omega)_0$, and delete the voxels in $Vox(\partial\Omega)$ that comply with an elimination criterion. To understand the implemented thinning algorithm, it is necessary to describe: (1) The neighbours of a voxel, (2) the simple point criterion, and (3) the end point criterion.

The neighbours of a voxel $p \in \Omega$ are the 26 voxels that surround such a voxel. This neighborhood is sub-classified as 6- 18- and 26- neighbors, as follows: (a) 6-neighbor voxels share one face with the central voxel p . (b) 18-neighbor voxels share at least one edge with the central voxel p . (c) 26-neighbor voxels share at least one vertex with the central voxel [17]. In order to preserve the connectivity of the skeleton, the simple point criterion is implemented [15,17]. The simple point criterion, requires a voxel p to satisfy the following conditions:

1. The central voxel p has a white voxel in its 6-neighbor.
2. The central voxel p has a black voxel in its 26-neighbor.
3. The set of black 26-neighbors of the central voxel p is 26-connected.
4. The set of white 6-neighbors of the central voxel p is 6-connected in the set of white 18-neighbors.

On the other hand, the end point criterion is implemented to preserve the shape of the input body. A voxel satisfies the end point criterion if it has a single voxel neighbor in its 26-neighbor set.

The implemented thinning algorithm follows the next steps: (1) To voxelize the input mesh $M = (X, T)$ generating the voxel set $Vox(\Omega)_0$. (2) The boundary voxels $Vox(\partial\Omega)_k$ are detected by evaluating the simple point and the end point criteria. (3) The voxel set is compared with the boundary voxel set. If the voxel set is equal to the boundary ($Vox(\Omega)_k = Vox(\partial\Omega)_k$), the thinning procedure finishes. (4) the random labelling $Rand(Vox(\Omega)_k)$ is applied. (5) The voxels in the skin $\partial\Omega$ that comply with the simple point criterion are removed. Finally, (6) steps 2-5 are repeated until step 4 is satisfied, generating the resulting skeleton $SK(\Omega)$.

3.2 Volume estimate

To retrieve the original body Ω from the skeleton $SK(\Omega)$, it is necessary to estimate the distance of each voxel in $SK(\Omega)$ to the boundary $\partial\Omega$. For this purpose, the next two methods can be used:

1. Arcelli et. al. [2] use the method of reverse distance transform to retrieve the original region. This method find the radius inscribing the sphere with the minimum radius that is tangent to $\partial\Omega$ in one or more points.
2. Munoz et. al. [18] use a measure of mass (Computed using the diffusive advection proposed by Jalba et. al. [11]) in each voxel of $SK(\Omega)$ to approximate a cylinder in each segment and considering a constant density.

3.3 Graph skeleton representation

To generate the spectral representation, it is necessary to convert the voxelized skeleton $SK(\Omega)$ to its graph representation $G = (V, E)$, where G is a connected and non-oriented graph defined by a set of vertices V and a set of edges E . In graph representation, each point $p_i \in V$ is represented by the center point of each voxel in $SK(\Omega)$, and each edge $e_i \in E$ is computed by connecting the adjacent black-voxels in each 26-neighborhood. A function $f_i: int(e_i) \rightarrow R$ is assigned to each edge $e_i \in E$. f_i stores the radius of every point contained in the line segment $int(e_i)$, defined by the edge e_i . In this process, any 3-vertex closed cycle (triangle) is deleted and replaced with its respective barycenter.

3.4 Spectral segmentation

The spectral decomposition of the graph $G = (V, E)$ is a representation of the skeleton $SK(\Omega)$ in the frequency domain. As such, this spectral decomposition preserves the topology of G . However, the low frequencies of this skeleton can be extracted from the first 3 eigenvectors (i.e. with lowest non-zero eigenvalue) of the Laplacian operator on graphs [24,28]. The graph Laplacian L and its corresponding spectrum is computed as follows:

1. Find the $n \times n$ weighted Laplacian matrix L of G , defined as follows:

$$L_{ij} = \begin{cases} -\sum_{v_k \in N_i} L_{ik} & \text{if } i = j \\ -\frac{1}{\|v_i - v_j\|} & \text{if } (v_i, v_j) \in E \\ 0 & \text{if } (v_i, v_j) \notin E \end{cases} \quad (1)$$

2. Find the first 3 eigenvectors $(\lambda_1, \lambda_2, \lambda_3)$ of L , with smallest (non-zero) eigenvalues $(\lambda_1, \lambda_2, \lambda_3)$. That is to say, the eigenpairs $(\lambda_i > 0, \mu_i \in R^n)$ that satisfy the following equation:

$$L\mu = \lambda_k \mu_k \quad (2)$$

With the lowest frequencies of $G = (V, E)$ identified, a manual segmentation is applied generating $G_S = (G_1, G_2, \dots)$ that represent the large features of $SK(\Omega)$. Each G_i with $i = \{1, 2, \dots\}$ is defined by the extreme vertices of the large features.

3.5 Small death branches removal

With the identification of the large features $G = (G_1, G_2, \dots)$, the next step is use the shortest path algorithm of Dijkstra [8] to remove the small death branches in $G = (V, E)$. This process compute a new graph $G_L = (V_L, E_L)$ without the small death branches where $G_L \subset G$, the vertices $V_L \subset V$ and the edges $E_L \subset E$.

3.6 High frequencies removal

To reduce the oscillations in the skeleton, a maximal excursion algorithm (edge linearization) is used [18]. This reduce the high frequencies and improve the smoothness of $G_L = (V_L, E_L)$, generating a smoothed graph $G_M = (V_M, E_M)$. In each iteration of this algorithm, is evaluated if the maximum distance of each new segment is greater than a predefined threshold. The algorithm stops when the maximum distance of all new segments is less than the threshold. The mass of a new segment is the sum of the mass of all the vertices in the new segment created by the old skeleton.

3.7 Retrieval of the original region

To retrieve the original body Ω , an alpha-shape based inversion is used [9], generating a 2-manifold triangular mesh $M_L = (X_L, T_L)$ that is an estimation to $\partial\Omega$, defined by a set of points $X_L \subset \partial\Omega$ and a set of triangles T_L . For this propose, a set of discs with different radius are created in each segment of $G_L = (V_L, E_L)$. The points of these discs are the input of the alpha-shape.

4. Results

The boundary representation $M = (X, T)$ models were voxelized to find the medial axis. Figure 2 shows the models *Truss*, *Snake* (Extracted from [26]) and *Horse* (Extracted from [5]). Figures 3(a), 3(b) and 3(c) shows the voxel skeletons $SK(\Omega)$ with their respective mass distribution. Figures 3(d), 3(e) and 3(f) shows graph skeletons with the set of computed radius using the reverse distance transform method. In this step $SK(\Omega)$ presents high frequencies (e.g. Figure 3(b)) and small death branches (Figure 3(c)).

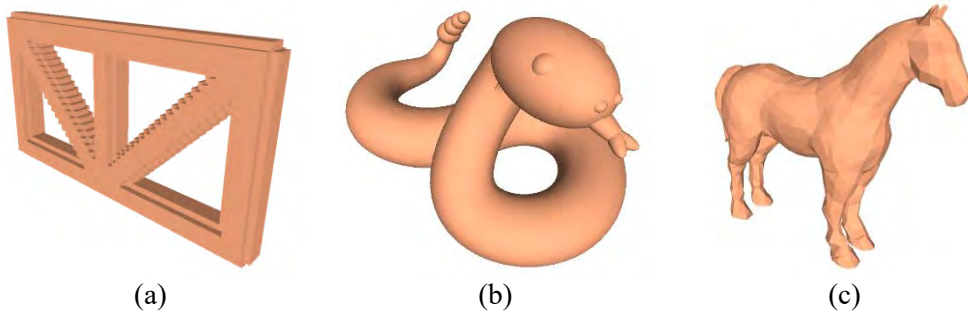


Figure 2. Boundary representation $M = (X, T)$ of the test datasets. (a) *Truss* dataset. (b) *Snake* dataset. (c) *Horse* dataset.

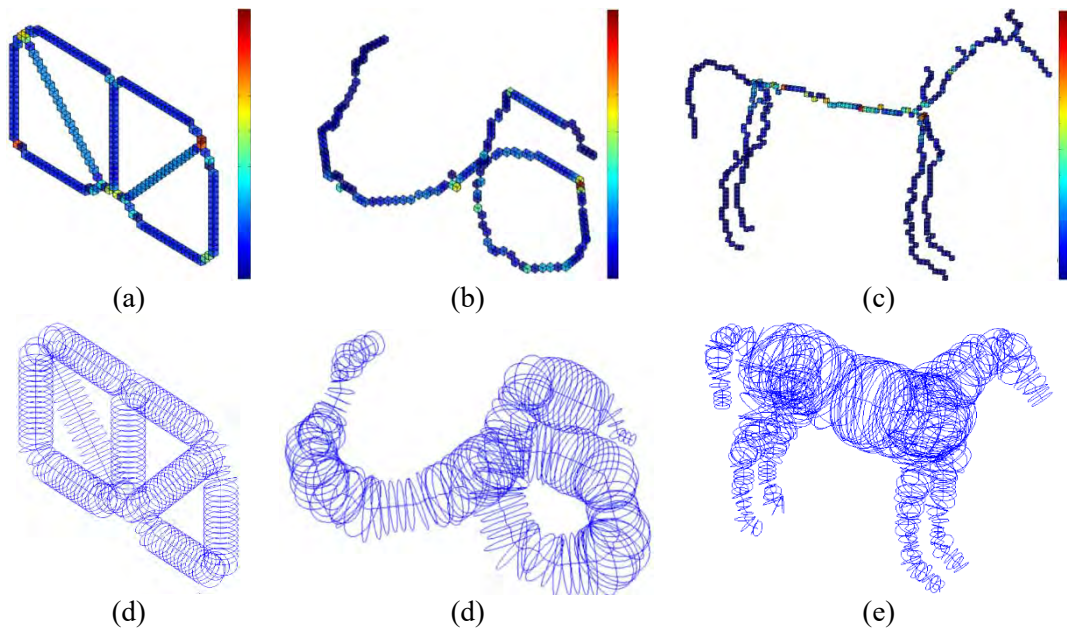


Figure 3. Resulting skeletons from our unskewed thinning algorithm and the respective node masses. The skeletons present high frequencies and short dead branches. (a) *Truss* (b) *Snake* and (c) *Horse* skeleton with the distribution of mass. (d) *Truss* (e) *Snake* and (f) *Horse* graph skeleton with the computed radius.

Figure 4(a) shows the skewed *Horse* graph skeleton produced using the algorithm without the proposed randomization. Figure 4(b) shows the unskewed *Horse* graph skeleton produced using the algorithm with the randomization. The comparison between the red squares in Figures 4(a) and 4(b) shows that the small death branches are skewed in Figure 4(a) due the direction of analysis, while the branches are unskewed in the randomized version (our algorithm).

Figures 5(a) and 5(b) shows the skewed and unskewed retrieved model *Snake* (beige color) computed using the reverse distance transform method and compared with the input mesh $M = (V, E)$ (blue color). Compared with the Figure 5(b) (unskewed), Figure 5(a) (skewed) have significant less volume that the original body, indicating a deviation grater from the medial axis.

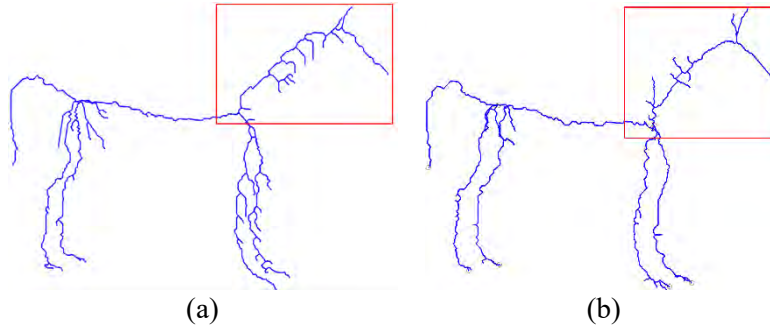


Figure 4. Comparison between skewed and unskewed graph skeletons.

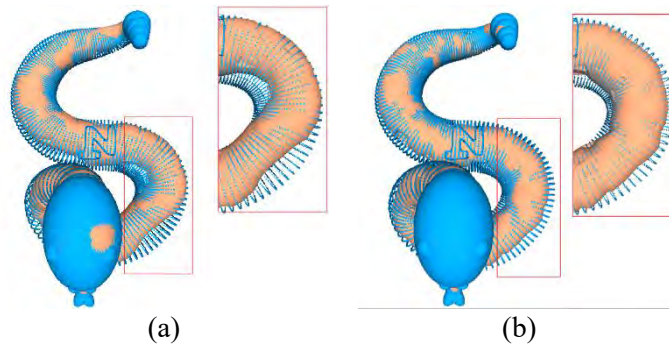


Figure 5. Comparison retrieval models (Beige color) and the original region (Blue color).

Figure 6(a) shows the graph skeleton $G = (V, E)$ of the *Horse* dataset with the small death branches and high frequencies. Figure 6(b) shows the eigenvalues with the lowest frequencies of L where the small death branches are squashed and allows the manual graph segmentation $G = (G_1, G_2, \dots)$. Figure 6(c) shows $G = (V, E)$ after the application of the shortest path algorithm using the $G_s = (G_1, G_2, \dots)$, that deletes the small death branches. Figure 6(d) shows $G = (V, E)$ after the application of the maximal excursion algorithm that reduces the high frequencies. The qualitative comparison shows that the final model (Figure 6(c)) is smoothed in comparison with the initial graph skeleton (Figure 6(a)). Also, the number of points that represent the model decrease significantly, from 735 points (Figure 6(a)) to 25 points (Figure 6(d)).

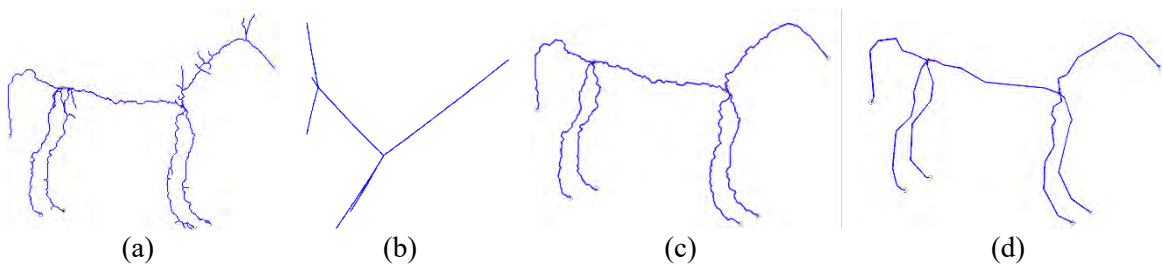


Figure 6. Geometric simplification of the graph skeleton. (a) *Horse* graph skeleton with small death branches and high frequencies. (b) Lowest frequency eigenfunction of *Horse* graph. (c) *Horse* graph skeleton after deleting small death branches. (d) *Horse* graph skeleton after reducing high frequencies.

Figure 7 shows the retrieval model using an alpha shape-based inversion. Figures 7(a), 7(b) and 7(c) shows the retrieval after the deletion of the small death branches that is a good approximation to Ω .

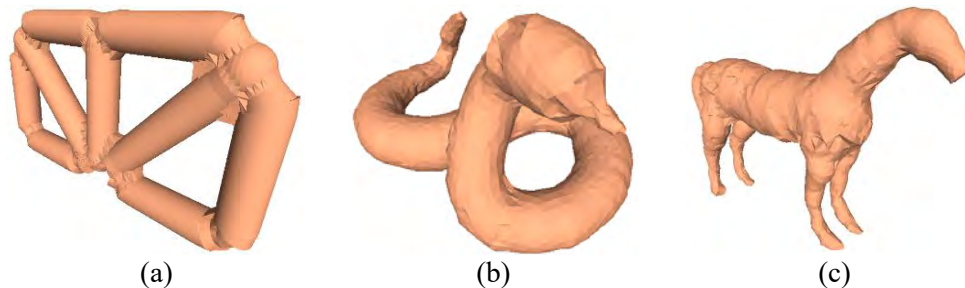


Figure 7. Retrieval of the original model using alpha-shape based inversion. Retrieval of the model (a) *Truss* (b) *Snake* and (c) *Horse*.

5. Conclusions and future work

This manuscript presents a novel method to improve the centeredness centrality and the smoothness of the skeletons generated by thinning algorithms. The presented method articulates: (a) a random labelling applied to the voxel-based algorithm that neutralize the skewed produced by the order of analysis in the skeletons, (b) the deletion of the small death branches using a spectral segmentation, and (c) the smoothing of the skeleton reducing the high frequencies with a maximal excursion algorithm (edge linearization). The results show that using the spectral segmentation in graph of skeleton is an effective method to produce a smoothed skeleton and delete undesirable noise. Also, the retrieved models show that our method can produce a good approximation of the original body. Future work includes (1) the tune the spectral filtering of high frequencies off the basic skeleton and (b) extend the algorithm to solid regions whose skeletons mix 1D and 2D entities.

References

- [1] Adam A 2020 *Mesh voxelisation*. [Online]. Available from: <https://www.mathworks.com/matlabcentral/fileexchange/27390-mesh-voxelisation>.
- [2] Arcelli C, di Baja G S and Serino L 2010 *Distance-driven skeletonization in voxel images (Transactions on Pattern Analysis and Machine Intelligence vol 33(4))* (IEEE) pp 709-720.
- [3] Bucksch A and Lindenbergh R 2008 *CAMPINO—A skeletonization method for point cloud processing (ISPRS journal of photogrammetry and remote sensing vol 63(1))* pp 115-127.
- [4] Cao T T, Tang K, Mohamed A and Tan T S 2010 *Parallel banding algorithm to compute exact distance transform with the GPU* pp. 83-90
- [5] Chemlife 2019 *Low poly horse* [Online]. Available from: <https://www.thingiverse.com/thing:925638>.
- [6] Couprie M and Bertrand G 2016 *Asymmetric parallel 3D thinning scheme and algorithms based on isthmuses (Pattern Recognition Letters vol 76)* pp 22-31.
- [7] Dey T K and Zhao W 2002 *Approximate medial axis as a voronoi subcomplex (In Proceedings of the seventh ACM symposium on Solid modeling and applications)* pp. 356-366
- [8] Dijkstra E W *A note on two problems in connexion with graphs Numerische Mathematik* 1959 pp 269–271
- [9] Edelsbrunner H and Mücke E P 1994 *Three-dimensional alpha shapes. (ACM Transactions on Graphics (TOG), 13(1))* pp 43-72.
- [10] Foskey M, Lin M C and Manocha D 2003 *Efficient computation of a simplified medial axis (J. Comput. Inf. Sci. Eng. 3(4))* pp 274-284.
- [11] Jalba A C, Sobiecki A and Telea A C 2015 *An unified multiscale framework for planar, surface, and curve skeletonization (IEEE transactions on pattern analysis and machine intelligence 38(1))* pp 30-45

- [12] Jin X and Kim J 2017 *A 3D skeletonization algorithm for 3D mesh models using a partial parallel 3D thinning algorithm and 3D skeleton correcting algorithm (Applied Sciences vol 7(2))* p 139.
- [13] Liu L, Chambers E, Letscher D and Ju T 2010 *A simple and robust thinning algorithm on cell complexes (In Computer Graphics Forum vol 29)* (Wiley Online Library) pp 2253–2260.
- [14] Livesu M, Guggeri F and Scateni R 2012 *Reconstructing the curve-skeletons of 3d shapes using the visual hull. (IEEE transactions on visualization and computer graphics vol 18(11))* pp 1891-1901.
- [15] Ma C M 1994 *On topology preservation in 3D thinning (CVGIP: Image understanding vol 59(3))* pp 328-339
- [16] Ma J, Bae S W and Choi S 2012 *3D medial axis point approximation using nearest neighbours and the normal field (The Visual Computer vol 28(1))* pp 7-19.
- [17] Malandain G and Bertrand G 1992 *Fast characterization of 3D simple points (In 11th IAPR International Conference on Pattern Recognition Vol III Conference C: Image, Speech and Signal Analysis Vol 1)* (IEEE Computer Society) pp 232-235
- [18] Munoz J M, Ruiz-Salguero O, Montoya-Zapata D, Cortés C and Cadavid C A 2018 *Direct Scalar Field-to-Truss Representation and Stress Simulation of Open Pore Domains (In STAG)* pp 25-34
- [19] Saha P K, Borgfors G and di Baja G S 2016 *A survey on skeletonization algorithms and their applications (Pattern recognition letters vol 76)* pp 3-12
- [20] Saha P K, Borgfors G and di Baja G S 2017 *Skeletonization and its applications—a review (Skeletonization)* (Academic Press) pp 3-42
- [21] Siddiqi K, Bouix S, Tannenbaum A and Zucker S W 2002 *Hamilton-jacobi skeletons (International Journal of Computer Vision vol 48(3))* pp 215-231
- [22] Sobiecki A, Jalba A and Telea A 2014 *Comparison of curve and surface skeletonization methods for voxel shapes (Pattern Recognition Letters vol 47)* pp 147-156
- [23] Song C, Pang Z, Jing X and Xiao C 2018 *Distance field guided L_1 -median skeleton extraction (The Visual Computer vol 34(2))* pp 243-255
- [24] Stanković L, Daković M and Sejdić E 2019 *Introduction to graph signal processing (In Vertex-Frequency Analysis of Graph Signals)* (Cham: Springer) pp 3-108
- [25] Tagliasacchi A, Alhashim I, Olson M and Zhang H 2012 *Mean curvature skeletons (In Computer Graphics Forum vol 31(5))* (Oxford UK: Blackwell Publishing Ltd.) pp. 1735-1744
- [26] NT Toys. 2019 *Snake (nt animals)*. [Online]. Available from: <https://www.thingiverse.com/thing:1220259>
- [27] Wu J, Duan H and Zhong Q 2011 *3D image skeleton algorithms (In 2011 IEEE International Conference on Anti-Counterfeiting, Security and Identification)* (IEEE) pp. 97-100
- [28] Zhang H, Van Kaick O and Dyer R 2010 *Spectral mesh processing (In Computer graphics forum Vol 29(6))* (Oxford, UK: Blackwell Publishing Ltd.) pp 1865-1894