An Adaptive Hierarchical Approach to the Extraction of High Resolution Medial Surfaces

Luca Rossi, Andrea Torsello
Università Ca' Foscari - Venice, Italy
`lurossi@dsi.unive.it, torsello@dais.unive.it`

Abstract

We introduce a novel algorithm for medial surfaces extraction that is based on the density-corrected Hamiltonian analysis [20]. The approach extracts the skeleton directly from a triangulated mesh and adopts an adaptive octree-based approach in which only skeletal voxels are refined to a lower level of the hierarchy, resulting in robust and accurate skeletons at extremely high resolution. The quality of the extracted medial surfaces is confirmed by an extensive set of experiments.

1. Introduction

The skeleton has proven to be a valuable and widely used shape descriptor for a number of tasks such as 2-D and 3-D shape recognition [13, 19], volumetric models deformation [21], segmentation [15] and protein structure identification [1]. When working in two dimensions, the skeleton, or medial axis transform, is defined as the locus of the centers of the maximal inscribed circles bitangent to the shape boundary. The interest in this descriptor stems from its being a concise representation of the original shape, which is topologically equivalent to it, and invariant to several shape deformations.

Broadly speaking, the algorithms to extract a skeleton can be divided into four main categories. Approaches in the first class simulate Blum's grassfire transform by iteratively eroding layers from the shape [5]. These algorithms, while fast and simple to implement, are sensitive to Euclidean transformations. Approaches in the second class detect the ridges of the distance map either directly [7] or by evolving a series of curves, such as snakes, under a potential energy field defined by the distance map [10]. Although these methods fulfill the geometrical constraint, the topological correctness is not guaranteed. Approaches in the third class are based on the Voronoi diagram of a triangulated shape [12]. These techniques extract the correct topology, and are invariant under Euclidean transformations, but

extract very noisy skeletons. The fourth, and final, class of methods is based on the analysis of the differential structure of the boundary. In [9], the boundary is segmented at points of maximal curvature and the skeleton is computed through the Voronoi diagram of these segments. Leymarie and Leving [10] proposed a similar approach based on active contours, where the curve is evolved according to the eikonal equation, and the points where the wavefront collapses define the skeletal points. Unfortunately, as in [9] this requires an initial segmentation of the boundary at curvature extrema, which is itself a challenging problem.

Another important method that belongs to this class stems from the Hamiltonian analysis of the boundary flow dynamics [18]. Siddiqi et al. state that the singular points where the system ceases to be Hamiltonian (i.e., an energy conservation principle is violated) are responsible for the formation of skeletal points. Torsello and Hancock [20] improve the localization properties of the approach by taking into account the effects of the boundary curvature.

Today, the wide availability of cheap 3D scanning devices renders topical the automated extraction of a representation which provides a simple venue to perform shape analysis and representation under deformation and articulation. For this reason, the design of efficient algorithms for 3D skeleton extraction is of pivotal importance. Note also that while in 2D the image needs to be segmented in order to extract the shapes, in 3D the objects are naturally modeled as distinct meshes, thus rendering the skeletonization much more practical. The addition of a third dimension, however, renders the task of medial surfaces extraction particularly challenging. First, there is an exponential growth of the number of voxels, which may render the computation impracticable when a high resolution is needed. Further, volumetric objects are commonly represented as triangle meshes, that may eventually need to be voxelized before any further computation is done. Depending on the resolution chosen, this discretization might yield the wrong topology. Moreover, not only the spatial and time complexity of the algorithm is increased, but also tasks that are almost trivial in two dimensions, such as ensuring the topological equiv-

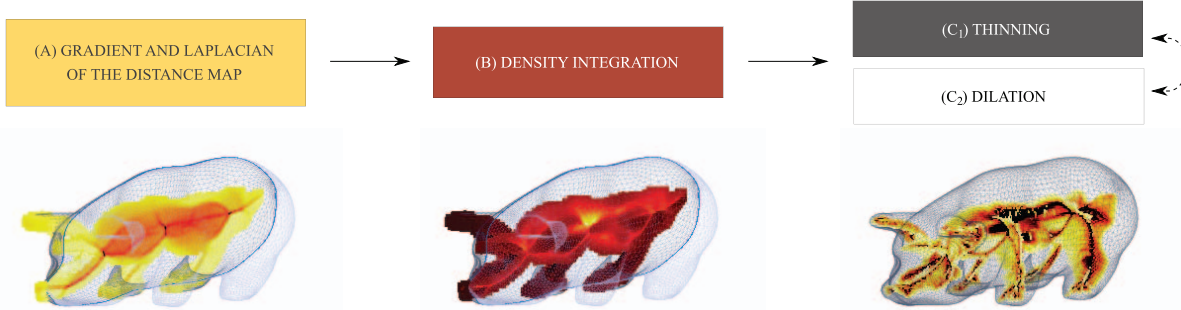| (A) GRADIENT AND LAPLACIAN OF THE DISTANCE MAP | (B) DENSITY INTEGRATION | (C_1) THINNING |
| | | (C_2) DILATION |

Figure 1. Steps to refine the skeleton: a) computation of the gradient and Laplacian of the distance map; b) integration of the log-density in the voxels with a full neighborhood; c) alternating thinning and dilation step to detect skeletal voxels at the current level of the octree.

alence between the object and its skeleton, become more challenging when a third dimension is added.

Note that in the literature there are two competing 3D generalizations of the skeleton: the curve (or line) skeleton [3], which provides a minimal representation for shape analysis and recognition, and the medial surfaces, which, on the other hand, carry enough information to recover the original shape. Moreover the medial surface is topologically equivalent to the shape in the sense that there exists a homotety that maps its segments (considered as two oriented surfaces) to the original mesh. The same is not tur of the line skeleton which is a lossy simplification of the shape. Finally, the curve skeleton is ill-defined in some degenerate cases, as for example the shape of a cup. In this paper, we therefore concentrate on the extraction of medial surfaces from triangulated meshes.

Arcelli et al. [2] propose a distance-driven algorithm that is topology preserving but works only on voxelized objects, and thus is unable to cope with high resolution inputs. The same holds for the algorithm proposed by Siddiqi et al. [19], which is a generalization to three dimensions of the Hamilton-Jacobi skeleton. Again, this approach is topology preserving but requires a discretized input and moreover it suffers from the same limitations of its two-dimensional counterpart, since it doesn't take into account the effects of boundary curvature. A more robust algorithm is that of Reniers et al. [16], where both the curve and the surface skeletons are located by means of an advection-based importance measure. Unfortunately this measure turns out to be well defined only for genus 0 shapes.

Another approach is that of Bai et al. [4] and Quadros et al. [14], who propose to use adaptive octrees in order to reduce the spatial and time complexity. This allows some parts to be discretized more densely while the rest can be analyzed at a coarser scale. However, both these approaches work on a precomputed octree, where the grid refinement criterion is based on simple heuristics.

On the other hand Yoshizawa et al. [21] take a Voronoi-based approach, where the skeleton of a mesh is approxi-

mated by a skeletal mesh having the same connectivity as the original mesh. These approaches are fast and do not require an initial voxelization, but extract only an approximation of the skeleton and are extremely sensible to small perturbations of the boundary.

In this paper we introduce a novel algorithm for medial surfaces extraction that is based on a generalization to three dimensions of the density-corrected analysis of Torsello and Hancock [20], while taking an adaptive octree-based approach for the discretization of the initial mesh where we iteratively decide which voxels to refine based on the local value of the divergence of the momentum field, i.e., the confidence we have in that point being skeletal.

2. Hamilton-Jacobi Skeleton

Let the distance map D be a function that assigns to each point in the interior of the shape its distance to the closest point on the object boundary $\mathbf{B}$. We define the velocity field $\mathbf{F} = \nabla D$. In their original formulation, Siddiqi et al. propose to label as skeletal those points in which the divergence of $\mathbf{F}$ is non-zero, under the assumption that the field $\mathbf{F}$ is conservative everywhere except on the skeleton [18]. However, under a compressing front, the divergence can be negative also at non-skeletal locations. The problem was overcome with the introduction of the concept of normalized flux , however, due to the discrete structure of the lattice, the integration radius has a lower bound of one pixel. Since the divergence of the velocity field in $\mathbf{p}$ depends on the local boundary curvature, assuming an integration radius of one pixel, the value of the normalized flux at $\mathbf{p}$ will be $N\phi_A(\mathbf{F})(p) = -\frac{1}{2}k(\mathbf{p})$, where $k(\mathbf{p})$ is the curvature of the evolving boundary at $\mathbf{p}$. The problem is that near the endpoints of the skeleton the value of the curvature will tend to infinity, thus the discrete normalized flux diverges in their proximity.

A solution to this problem was proposed in [20] where the authors observe that the linear density of the boundary is not constant over time when the boundary is curved, so they propose to resort to a mass conservation principle. They

(a) $32 \times 32 \times 32$

(b) $64 \times 64 \times 64$

(c) $128 \times 128 \times 128$

(d) $256 \times 256 \times 256$

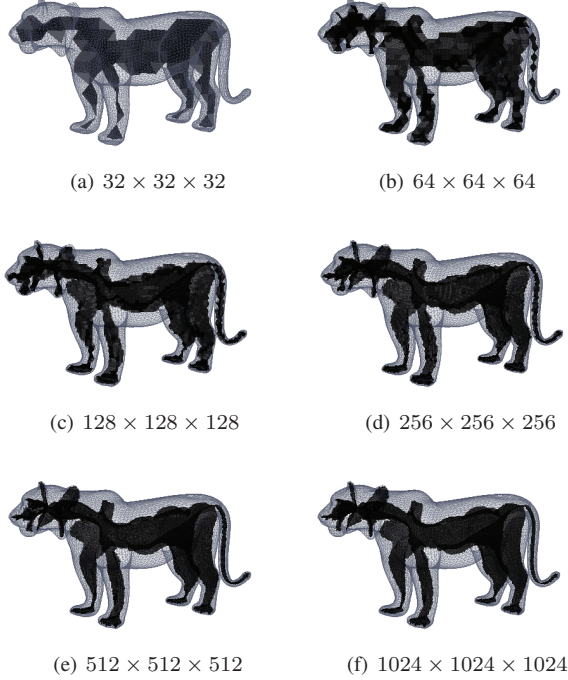(e) $512 \times 512 \times 512$

(f) $1024 \times 1024 \times 1024$

Figure 2. The hierarchical refinement of the medial surfaces. The skeletal points are meshed for ease of visualization.

state that it is the momentum field $\mathbf{M} = \rho\mathbf{F}$ that is conservative. Here ρ is a scalar field that assigns to each point the linear density of the inward-evolving boundary front. In other words, we have $\nabla\cdot(\rho\mathbf{F}) = 0$ at any non-skeletal point and thus also $\phi_A(\rho\mathbf{F}) = 0$ for any region A not containing a skeletal point.

Applying the rule of product differentiation to the conservation equation and setting $\sigma = \log(\rho)$ we obtain

$$\nabla\sigma \cdot \mathbf{F} = -\nabla \cdot \mathbf{F}, \tag{1}$$

which can be further reduced to the system of ordinary differential equations along the path of boundary points

$$\begin{cases} \frac{\partial}{\partial t}\sigma(s(t)) = -\nabla \cdot \mathbf{F}(s(t)) \\ \frac{\partial}{\partial t}s(t) = \mathbf{F}(s(t)) \end{cases} \tag{2}$$

where $s(t)$ is the trajectory of a boundary point under the eikonal equation.

3. Hierarchical Skeletonization

In this work we derive a hierarchical refinement approach which computes high resolution medial surfaces using a density-corrected Hamiltonian analysis. To this end, we start at an initial resolution res_{min} by constructing a complete octree of height $\log_8(res_{min})$, and we compute an initial medial surface through a 3D generalization of

the curvature corrected Hamiltonian analysis of Torsello and Hancock [20]. Once the initial approximation of the skeleton is to hand, we iteratively increase the resolution by subdividing the leaves of the octree with a large value of $\nabla\cdot(\rho\mathbf{F})$, i.e., the voxels that are most likely to contain skeletal points. The Hamiltonian analysis is then carried over the newly created octree level and the refinement process is iterated until the required resolution res_{max} and octree level $\log_8(res_{max})$ is reached.

In order to carry over the Hamiltonian analysis at a lower octree level the following steps must be undertaken (see Figure 1): a) computation of the distance map, its gradient, and its Laplacian at the newly refined voxels; b) integration of the front-density; c) thinning and dilation.

In the final step, the extraction of the skeleton is obtained through a thinning process, but then we need to perform a morphological dilation. We perform two different but consecutive dilations: The first immediately before subdividing skeletal points creating the new octree level which is done to guarantee that the exploded points have a complete neighborhood around each skeletal point. The second dilation is performed after the thinning process to partially compensate for discretization errors incurred at the coarser levels (see Section 3.4 for the details). With this high-level overview in mind, we will now present all the computational ingredients needed by the proposed approach.

3.1. Distance, gradient and Laplacian

We assume we can compute the distance map at each leaf of the current octree. In our implementation we perform distance queries against an Axis Aligned Bounding Box (AABB) tree holding the original mesh, but other, more efficient approaches can be adopted. Once the distance map is to hand, its gradient and divergence can be determined. Note, however, that while in the beginning all the leaves of the octree are at the same level and thus the gradient and the Laplacian can be approximated using the finite difference method, as the skeleton is refined there will be several voxels at different levels of resolution. For this reason we need to resort to a different approximation method that is able to cope with a non-uniform grid setting.

We compute the gradient by fitting a hyperplane over all the neighbors of $\mathbf{x}$. More formally, given a set of points $\{(x_i, y_i, z_i, d_i)\}_{i=1}^m$, where $(x_i, y_i, z_i)^T$ is a neighbor of $\mathbf{x}$ and d_i its distance to the boundary, we look for the coefficients A, B, C, D so that the hyperplane $d = Ax + By + Cz + D$ best fits the samples in a least squares sense. The gradient is then $\mathbf{F}(\mathbf{x}) = \frac{(A,B,C)^T}{||(A,B,C)^T||}$.

As for the Laplacian of the distance map, i.e., the divergence of the velocity field, we compute its value using the following discretization of the divergence theorem:

$$\nabla \cdot \mathbf{F}(\mathbf{x}) = \lim_{|U|\to 0} \frac{\int_{\delta U} \mathbf{F}(s) \cdot \mathbf{n}(s)\, ds}{|U|} \approx$$

$$\frac{\sum_{t=1}^{8} \frac{1}{3} A_t \mathbf{n}_t \cdot \left(\sum_{\mathbf{p}\in V_t} \mathbf{F}(\mathbf{p}) \right)}{|U_0|} \qquad (3)$$

where U_0 is the convex hull of the 6-neighbors of $\mathbf{x}$ and A_t, $\mathbf{n}_t$, and V_t are respectively the area, the normal, and the set of vertices of the (triangular) faces of U_0.

3.2. Integration of the momentum field

Once the distance, gradient and Laplacian have been computed, we can integrate the density in the newly subdivided skeletal points.

The keypoint of the density integration is that it is carried out only on points that have a full 27-neighborhood, the voxels with a non-homogeneous neighborhood simply inherit the value of the density and divergence fields of their parent node. The reason for this is that an inhomogeneous neighborhood induces a higher discretization error to the direction of the gradient which will severely affect the accuracy of the integration step. In order to guarantee that all the children of the skeletal voxels at the previous octree level will indeed have a complete neighborhood, we perform a dilation of the skeleton at the higher level before subdividing it to form the next level. In this way the voxels with non-homogeneous neighborhood will form a 1-voxel thick boundary at the current octree level and will be children of the dilation voxels rather than of the skeletal voxels. This dilation can simply be considered a part of the last thinning/dilation step of the refinement of the previous level, which will be described later.

The computation of the momentum field is performed by solving equation 2. Note however that since the skeleton is defined as the set of singularities of the momentum field, we expect the density field to have different values at opposite sides of a medial surface, thus any finite difference solution must not cross a skeletal surface. We follow [20] and integrate the equation in the time domain, thus ensuring that the integration of the log-density σ only references values of σ calculated at points already crossed by the inward-evolving boundary.

Assume that there exists a family of surfaces $\mathbf{B}_t$ representing the inward evolution of the boundary $\mathbf{B}$, that can be locally parametrized as $\mathbf{B}_t(u, v)$ around any point $\mathbf{x}$. Then, we have

$$\sigma(\mathbf{B}_t(u, v)) = \sigma(\mathbf{B}_{t-1}(u, v)) +$$

$$\frac{1}{2}[\nabla \cdot \mathbf{F}(\mathbf{B}_t(u, v)) + \nabla \cdot \mathbf{F}(\mathbf{B}_{t-1}(u, v))] \qquad (4)$$

In the spatial domain, if $\mathbf{x} = \mathbf{B}_t(u, v)$ we have

$\mathbf{B}_{t-1}(u, v) \approx \mathbf{x} - \mathbf{F}(\mathbf{x})$, which, substituted into equation 4, yields

$$\sigma(\mathbf{x}) = \sigma(\mathbf{x} - \mathbf{F}(\mathbf{x})) + \frac{1}{2}[\nabla \cdot \mathbf{F}(\mathbf{x}) + \nabla \cdot \mathbf{F}(\mathbf{x} - \mathbf{F}(\mathbf{x}))] \qquad (5)$$

By interpolating $\sigma(\mathbf{x} - \mathbf{F}(\mathbf{x}))$ using trilinear interpolation, we can rewrite equation 5 as

$$\sigma(\mathbf{x})(1 - (1 - |F_1|)(1 - |F_2|)(1 - |F_3|)) =$$
$$\sigma(\mathbf{x} - \mathbf{F}(\mathbf{x})) - (1 - |F_1|)(1 - |F_2|)(1 - |F_3|)\sigma(\mathbf{x})$$
$$+ \frac{1}{2}[\nabla \cdot \mathbf{F}(\mathbf{x}) + \nabla \cdot \mathbf{F}(\mathbf{x} - \mathbf{F}(\mathbf{x}))] \qquad (6)$$

where, F_1, F_2, and F_3, are the three components of $\mathbf{F}(\mathbf{x})$ and, due to the fact that we use trilinear interpolation, $\sigma(\mathbf{x} - \mathbf{F}(\mathbf{x})) - (1 - |F_1|)(1 - |F_2|)(1 - |F_3|)\sigma(\mathbf{x})$ does not depend on the value of $\sigma(\mathbf{x})$.

Given this formulation, we can integrate the value of the log-density over the interior of the shape, starting from the most external voxels inwards. At the first level the most external voxels will be the boundary boxes, which have a unit density, and thus a null log-density. At all other steps, the external voxels will be the voxels with irregular neighborhood that inherit the log-density from their parents. Once the log-density has been integrated, we can proceed to compute the divergence of the momentum field in each point of the interior of the shape. The value of $\nabla \cdot (\rho \mathbf{F})(\mathbf{x})$ is given by approximating equation 1 as follows:

$$\nabla \cdot (\rho \mathbf{F})(\mathbf{x}) = \Delta \sigma e^{\sigma(\mathbf{x}) - \frac{1}{2}\Delta\sigma}$$
$$+ \frac{1}{2}\left[\nabla \cdot \mathbf{F}(\mathbf{x} - \mathbf{F}(\mathbf{x}))e^{\sigma(\mathbf{x} - \mathbf{F}(\mathbf{x}))} + \nabla \cdot \mathbf{F}(\mathbf{x})e^{\sigma(\mathbf{x})}\right] \qquad (7)$$

where $\Delta\sigma = \sigma(\mathbf{x}) - \sigma(\mathbf{x} - \mathbf{F}(\mathbf{x}))$. Note that, since the equations introduced in this section are to be evaluated at different levels of resolution, the integration step is actually dependent on the corresponding voxel size.

3.3. Divergence driven thinning

With the divergence information to hand, we can select the voxels that are likely to contain skeletal points and that will be further subdivided to form the next level in the octree. The skeleton extraction is based on a thinning process guided by the value of the divergence of the momentum field at each voxel.

We follow [19, 20] and thin the shape by iteratively removing boundary points that have a value of the divergence below a certain threshold. Note though that while we are guaranteed that the divergence is negative at skeletal points,
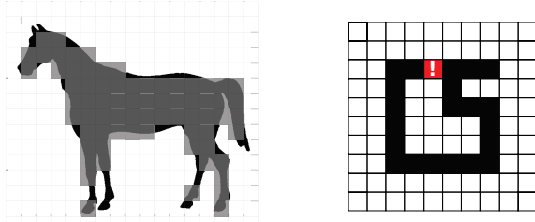
a) Coarse discretization levels might lose detail.

b) Dilation can change the topology of the skeleton.

Figure 3. Dilation is needed to regain detail lost at lower levels, but care must be given not to change the topology of the skeleton.

its value can become arbitrarily small, in particular it becomes zero on pure ligatures, i.e., skeletal segments generated by a boundary patch with a high negative curvature. Hence, without any further control on the thinning process we might actually end up altering the topology of the skeleton, introducing holes or even splitting it into disjoint parts. For this reason we need to remove only voxels whose removal will not change the topology of the skeleton.

Malandain et al. provide a way to determine the topological classification of a point $\mathbf{x}$ in a cubic lattice by computing two numbers: C^* and $\bar{C}$ [11].

Definition 1 C^* *is the number of the 26-connected components 26-adjacent to $\mathbf{x}$ in $B \cap N_{26}^*(\mathbf{x})$, where $N_n^*(\mathbf{x}) = N_n(\mathbf{x}) \setminus \mathbf{x}$ and B is the set of object points.*

Definition 2 $\bar{C}$ *is the number of the 6-connected components 6-adjacent to $\mathbf{x}$ in $W \cap N_{18}(\mathbf{x})$, where W is the set of background points.*

Malandain et al. [11] prove that $\mathbf{x}$ is simple (i.e., its removal will not change the topology of the shape) if and only if $C^*(\mathbf{x}) = 1$ and $\bar{C}(\mathbf{x}) = 1$.

With this result to hand, we can proceed with the thinning process by iteratively removing all simple points with a divergence above a given threshold. Due to the errors introduced by the discretization of the shape, after the first thinning process the medial surface can be two-voxel thick in certain regions. To ensure thinness at the highest resolution level we further thin the shape by removing all those points that are simple but not endpoints of the surface, regardless of their divergence. A necessary condition for a point to be an endpoint is to have three 6-adjacent background voxels.

3.4. Skeleton dilation

With the proposed hierarchical approach, once a voxel is flagged as non skeletal at any level, all its descendants will inherit the property. A problem with this is that fine details might be lost at coarser level, resulting in parts of the

(a) without dilation   (b) with dilation

Figure 4. Dilating the skeleton recovers details lost in the coarser levels.

skeleton that will be missing at all levels (see Figure 3 top). Further, note that the skeletal voxels detected at the coarsest level are not even guaranteed to be connected and, since all further processing is topology preserving, a disconnected skeleton will remain disconnected at all levels.

We address the latter problem by keeping only the largest component, while the missing detail is addressed by dilating the skeleton after it has been computed at each new level. This way, once the voxels are small enough to capture the detail, the skeleton will regrow into the missing parts.

Note that since the dilation adds new voxels to the current medial surface, we need to ensure that the topology is preserved, thus we dilate only into voxels that would become simple after the dilation (see Figure 3 bottom). Let I be the set of voxels that is fed as input of the thinning procedure. We mark the elements of the boundary of I as "border" points, then we thin the set as previously described.

After the thinning step, we check if some border voxel is still present and if so we dilate it. Then we compute D, $\mathbf{F}$, $\nabla \cdot \mathbf{F}$, ρ, $\nabla \cdot (\rho\mathbf{F})$ on the dilated set and we apply the thinning process again. The dilation-thinning process is iterated until the thinned skeleton contains no boundary voxels. This process gives us an adaptive dilation which adds only new candidate skeletal voxels with a large value of $\nabla \cdot (\rho\mathbf{F})$ and thus can be skeletal.

With this improvement, we are able to recover small details that might have been lost during the first discretizations, as well as longer skeletal segments. Figure 4 shows how critical this procedure is. The eagle model in the figure clearly needs a very dense voxelization in order to capture details such as the claws, or even entire parts such as the wings. With the proposed approach, one can simply start from a lower and less computationally intensive resolution and then refine the extracted skeleton to a certain desired resolution.

Finally, once the iterated dilation/thinning process gives us the final skeleton, we perform one final dilation step to ensure the presence of a complete 27-neighborhood around the new set of voxels on which we need to compute ρ and $\nabla \cdot (\rho\mathbf{F})$. At the last resolution level, the final dilation process is substituted with the endpoint-driven thinning that gives us a 1-voxel thick medial surface.

4. Experimental results

In this section we evaluate the quality of the proposed algorithm with a wide series of experiments. Here
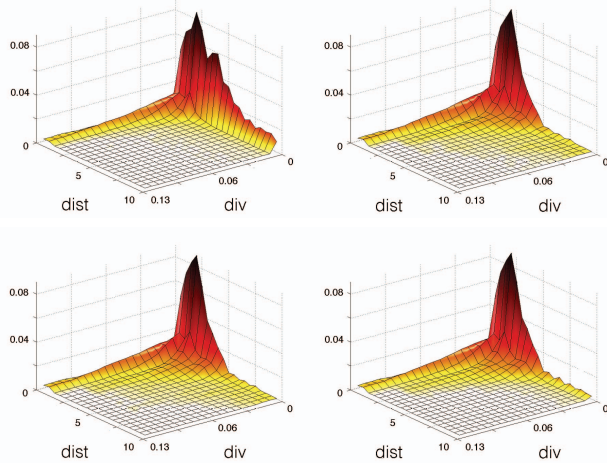
Figure 5. Distribution of the voxels as a function of both divergence and distance to the skeleton. The starting resolution ranges from $128 \times 128 \times 128$ to $16 \times 16 \times 16$, while the maximum resolution remains fixed at $128 \times 128 \times 128$. Note that the points with non-zero divergence are all located near the skeleton, while the points that are far from the skeleton have a value of the divergence equal to zero.

we present qualitative comparison with three different approaches, namely the Hamilton-Jacobi algorithm of Siddiqi et al. [18], the multiscale algorithm of Reniers et al. [16] and the Voronoi-based approach of Yoshizawa et al. [21]. Note that the first two methods work on a voxelized 3D shape, while the latter works directly on the mesh. The analysis has been performed on a selection of 40 shapes from the Princeton Shape Benchmark [17] and the SHREC 2010 database [6]. All skeletons are extracted with $res_{min} = 16$ and $res_{max} = 1024$, unless otherwise stated. Note that the proposed approach works independently of the shape's genus, and our dataset include shapes with genus greater than zero .

4.1. Skeleton localization

A common issue with the skeletonization algorithms present in the literature is the lack of quantitative evaluation against a ground truth, which is due to the difficulty of constructing a dataset with ground truth except for synthetic toy examples. Hence, in this section we try to address this shortcoming by measuring the approach's confusion by measuring the dispersion of skeleton indicator function, i.e., of the divergence information around the skeleton. In particular we evaluate the localization properties of the skeletons extracted with our algorithm and we compare it against the standard Hamilton-Jacobi approach.

To evaluate the localization properties of the density correction we plot the distribution of the voxels as a function of both divergence and distance to the skeleton. In order
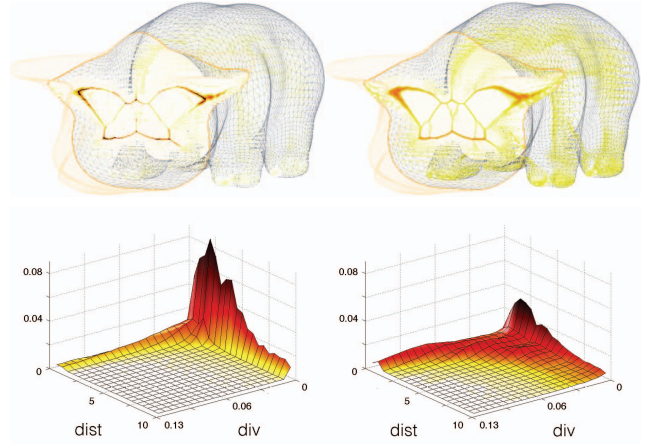

Figure 6. Comparison between the momentum field (left) and the velocity field (right). The left histogram shows a good localization of the skeleton, while in the right histogram we observe a non-negligible tail of distant points with non-zero divergence.

to evaluate the loss in localization caused by the hierarchical approach, we compare this distribution for shapes at the same target level but at different starting levels. In particular, the histograms in Figure 5 plot the average distribution of skeletons extracted at the maximum resolution of $128 \times 128 \times 128$, with starting resolutions going from $128 \times 128 \times 128$ (single level), to $16 \times 16 \times 16$ (multi-level (16)), thus all the skeletons were extracted with varying levels of hierarchical refinement.

First we note that when the hierarchical approach goes through more levels, the points tend to be more concentrated around the skeleton. This is to be expected since there is a decrease in the total number of voxels expanded. In general we see that the proposed algorithm yields a good localization of the skeleton, since the points with non-zero divergence are all located near the skeleton, while the points that are far from the skeleton have a value of the divergence equal to zero. However, we do observe a little noise due to the propagation of numerical errors, which is typical of hierarchical algorithms. Nonetheless, the distribution remains tightly peaked, with very few points far from the skeleton with a non-negligible divergence of the momentum field.

4.2. Comparison with other methods

Figure 6 compares the localization of the divergence of the momentum field against that of the velocity field as used by Siddiqi et al. [18]. As previously reported by Torsello and Hancock [20], even in 3D the momentum field localizes the skeleton much more tightly than the velocity field.

Here we show also a slice of the shape voxelization in order to reveal its interior, where the voxels are colored according to the value of the divergence, i.e., low values correspond to white while high (negative) values correspond to

Figure 7. Comparison of our approach (top left) against a standard Hamilton-Jacobi algorithm (top right), the Multiscale algorithm of Reniers et al. [16] (bottom left) and the Voronoi-Based approach of Yoshizawa et al. [21] (bottom right).
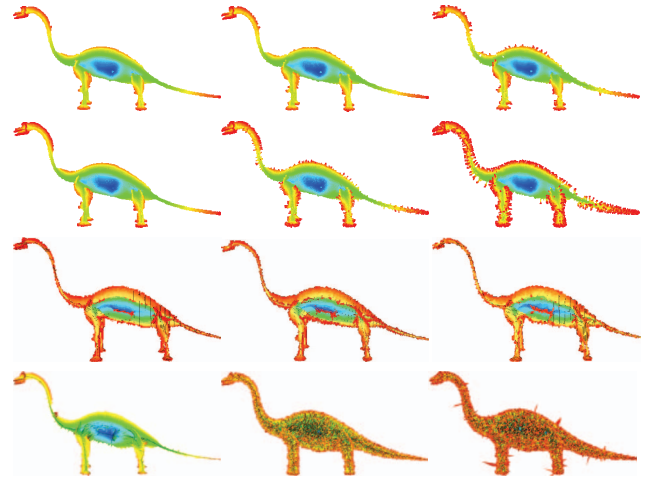

Figure 8. Effects of noise. The first column shows the skeletons extracted from the original object, while the second and the third columns show the skeletons after random vertex displacement of respectively 10% and 20% of the average edge applied to the shape From top to bottom: our approach, Hamilton-Jacobi, Multiscale [16] and Voronoi-based [21].

black. Recall that the value of $\nabla \cdot \mathbf{F}$ in a point p depends on the local boundary curvature and thus its value tends to infinity as p moves closer to a skeleton endpoint, even if p is not skeletal.

As a consequence of this, we observe some blurred areas around the endpoints of the medial surface. On the other hand, in the density-corrected slice we see a much sharper localization of the skeleton.

We also compare our algorithm with the Voronoi-Based approach of Yoshizawa et al. [21] and the Multiscale algorithm of Reniers et al. [16]. Both the implementations of [21] and [16] were downloaded from the authors websites, while we implemented the Hamilton-Jacobi algorithm simply by dropping the density integration procedure in our framework.

Figure 7 shows a qualitative comparison between the four methods. The Voronoi skeleton is clearly the noisiest one and in most cases fails to provide an acceptable approximation of the medial surface, although it is computationally significantly less expensive than the other algorithms. The Multiscale approach on the other hand performs quite well, although due to the complexity of processing a complete voxelization of the shape it was not able to reach the level of detail of our method. Finally, the Hamilton-Jacobi skeletons exhibit a few spurious skeletal segments due to the lack of the correction of the curvature effects.

A good skeletonization algorithm should also be able to deal with moderately noisy inputs. To this end, we approximate the skeletonization of the diffused shape by smoothing the distance map as in [20]. Hence, given a voxel and

its neighborhood, we update the local value of the distance by interpolating the values of the distance function on its neighbors [8].

Figure 8 shows the robustness to noise of the proposed approach. The results obtained by our algorithm and the Multiscale one are comparable. Note, though, that in the latter the robustness is achieved thanks to a fine tuning of the importance threshold, comes at the cost of losing some detail in the finer parts. On the other hand the Voronoi-based algorithm is unable to cope with the noise on the mesh boundary and thus performs much worse than the other approaches. Finally, the presence of noise clearly increases the formation of spurious branches in the Hamilton-Jacobi algorithm.

4.3. Time and space complexity

Perhaps the most obvious advantage of our algorithm is the decrease of space and time requirements. As for the theoretical complexity, it is governed by the sorting of points with respect to their distance to the boundary that takes place before the density integration, which is $O(n \log (n))$, where n is the number of leaves of the octree.

Figure 9 shows the average memory and time requirements of the proposed algorithm. Note that because of the higher memory requirements of the complete discretization, we couldn't start the discretization at resolutions higher than $256 \times 256 \times 256$. On the other hand, using the hierarchical approach we could easily reach much higher resolutions.
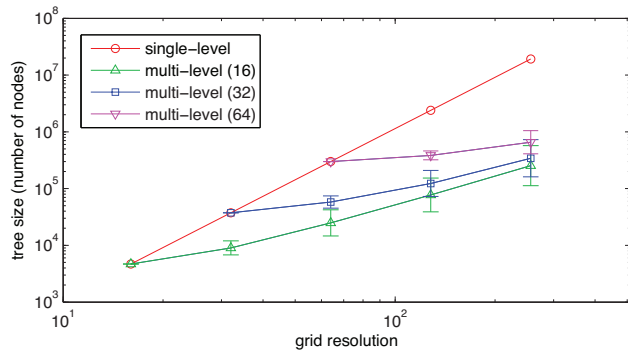
(a) memory requirement
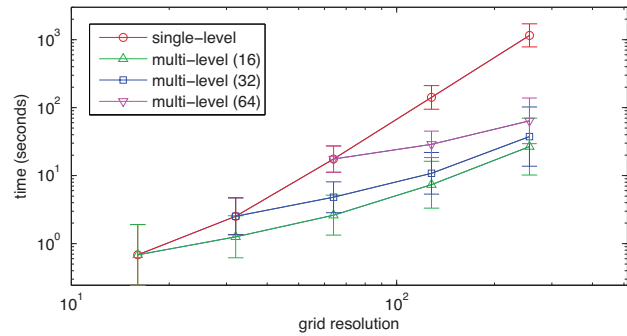
(b) execution time

Figure 9. The plots show the memory and time requirements for the computation of a series of skeleton with different levels of refinement. Our approach clearly outperforms the standard algorithm where the space is completely discretized.

5. Conclusions

In this paper we introduced a novel algorithm for medial surfaces extraction that is based on the density-corrected Hamiltonian analysis [20]. The approach extracts the skeleton directly from a triangulated mesh and adopts an adaptive octree-based approach in which only skeletal voxels are refined to a lower level of the hierarchy, resulting in robust and accurate skeletons at extremely high resolution. The quality of the extracted medial surfaces is confirmed by an extensive set of experiments.

References

[1] S. S. Abeysinghe, T. Ju, W. Chiu, and M. Baker. Shape modeling and matching in identifying protein structure from low-resolution images. In *Proc. ACM symposium on Solid and physical modeling*, pages 223–232, 2007. 1

[2] C. Arcelli, G. Sanniti di Baja, and L. Serino. Distance-driven skeletonization in voxel images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33:709–720, April 2011. 2

[3] O. K.-C. Au, C.-L. Tai, H.-K. Chu, D. Cohen-Or, and T.-Y. Lee. Skeleton extraction by mesh contraction. *ACM Trans. Graph.*, 27:44:1–44:10, August 2008. 2

[4] Y. Bai, X. Han, and J. L. Prince. Digital topology on adaptive octree grids. *J. Math. Imaging Vis.*, 34:165–184, 2009. 2

[5] G. Bertrand. A parallel thinning algorithm for medial surfaces. *Pattern Recogn. Lett.*, 16:979–986, 1995. 1

[6] A. Bronstein, M. Bronstein, U. Castellani, A. Dubrovina, L. Guibas, R. Horaud, R. Kimmel, D. Knossow, E. Von Lavante, D. Mateus, M. Ovsjanikov, and A. Sharma. SHREC 2010: Robust Correspondence Benchmark. In *Eurographics Workshop on 3D Object Retrieval (3DOR '10)*, 2010. 6

[7] S. Chang. Extracting skeletons from distance maps. *Int. J. of Comp. Sci. and Network Security*, 7(7), 2007. 1

[8] T. Ju, S. Schaefer, and J. Warren. Mean value coordinates for closed triangular meshes. In *ACM SIGGRAPH 2005 Papers*, SIGGRAPH '05, pages 561–566, New York, NY, USA, 2005. ACM. 7

[9] R. Kimmel, D. Shaked, N. Kiryati, and A. M. Bruckstein. Skeletonization via distance maps and level sets. *Comput. Vis. Image Underst.*, 62:382–391, November 1995. 1

[10] F. Leymarie and M. D. Levine. Simulating the grassfire transform using an active contour model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14:56–75, January 1992. 1

[11] G. Malandain, G. Betrand, and N. Ayache. Topological segmentation of discrete surfaces. *Int. J. Comput. Vision*, 10:183–197, April 1993. 5

[12] R. Ogniewicz and O. Kübler. Hierarchic voronoi skeletons. *Pattern Recognition*, 28(3):343–359, 1995. 1

[13] M. Pelillo, K. Siddiqi, and S. W. Zucker. Matching hierarchical structures using association graphs. In *Proc. 5th ECCV*, volume 2, pages 3–16. Springer-Verlag, 1998. 1

[14] W. R. Quadros, K. Shimada, and S. J. Owen. 3d discrete skeleton generation by wave propagation on pr-octree for finite element mesh sizing. In *Proc. 9th ACM symp. on Solid modeling and applications*, pages 327–332, 2004. 2

[15] D. Reniers and A. Telea. Skeleton-based hierarchical shape segmentation. In *Proc. IEEE Int. Conf. on Shape Modeling and Applications*, pages 179–188, 2007. 1

[16] D. Reniers, J. van Wijk, and A. Telea. Computing multi-scale curve and surface skeletons of genus 0 shapes using a global importance measure. *IEEE Transactions on Visualization and Computer Graphics*, 14:355–368, March 2008. 2, 6, 7

[17] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. In *Shape Modeling International*, 2004. 6

[18] K. Siddiqi, S. Bouix, A. Tannenbaum, and S. W. Zucker. The hamilton-jacobi skeleton. In *Proc. ICCV*, 1999. 1, 2, 6

[19] K. Siddiqi and S. Pizer. *Medial Representations: Mathematics, Algorithms and Applications*. Springer, 2008. 1, 2, 4

[20] A. Torsello and E. R. Hancock. Correcting curvature-density effects in the hamilton-jacobi skeleton. *IEEE Transactions on Image Processing*, 15(4):877–891, 2006. 1, 2, 3, 4, 6, 7, 8

[21] S. Yoshizawa, A. Belyaev, and H.-P. Seidel. Skeleton-based variational mesh deformations. In *Proc. EUROGRAPHICS*, volume 26, pages 255–264, 2007. 1, 2, 6, 7