Physics-Based Skeletons

Sylvain Bouix Pavel Dimitrov Carlos Phillips Kaleem Siddiqi

McGill University
School of Computer Science &
Center for Intelligent Machines
3480 University Street
Montreal, QC H3A 2A7, Canada
E-mail: {sbouix,pdimit,carlos,siddiqi}@cim.mcgill.ca

Abstract

Skeletons are widely used for 2D and 3D shape analysis in computer vision, bio-medicine and graphics. However, their numerical computation is non-trivial. Methods based on Voronoi techniques preserve the object's topology, but heuristic pruning measures are introduced to simplify the Voronoi graph. Approaches based on Euclidean distance functions can localize skeletal points accurately, but often at the cost of altering the object's topology. This paper presents an overview of an algorithm we have developed for computing 2D and 3D skeletons which addresses these concerns. The key idea is to measure the net outward flux of a vector field per unit area or volume and to detect locations where a conservation of energy principle is violated. This is done in conjunction with a thinning process applied in a discrete lattice. The method is robust and accurate, has low computational complexity, and preserves topology. We illustrate the approach with several computational examples.

1 Introduction

The 2D skeleton (medial axis) of a closed set $A \subset \mathcal{R}^2$ is the locus of centers of maximal open discs contained within the complement of the set [4]. An open disc is maximal if there exists no other open disc contained in the complement of A that properly contains the disc. The 3D skeleton (medial surface) of a closed set $A \subset \mathcal{R}^3$ is defined in an analogous fashion as the locus of centers of maximal open spheres contained in the complement of the set. Both types of skeletons have been widely used in bio-medicine for tasks involving object representation [19, 30], registration [13] and segmentation [24]. They have also been used for graph-based object recognition in computer vision [20, 33, 25, 14, 29], for animating objects in graphics [31, 22], and for manipulating them in computer-aided design. Despite their popularity, their numerical computation remains non-trivial. Most algorithms are not stable with respect to small perturbations of the boundary, and heuristic measures for simplification are introduced.

Approaches to computing skeletons can be broadly organized into three classes. First, methods based on *thinning* attempt to realize Blum's grassfire formulation [4] by peeling away layers from an object, while retaining special points [2, 11, 6, 18]. It is possible to define erosion rules in a lattice such that the topology of the object is preserved. However, these methods are quite sensitive to Euclidean transformations of the data and typically fail to localize skeletal points accurately. As a consequence, only a coarse approximation to the object is usually reconstructed [18, 3, 11].

Second, it has been shown that under appropriate smoothness conditions, the vertices of the Voronoi diagram of a set of boundary points converges to the exact skeleton as the sampling rate increases [23]. This property has been exploited to develop skeletonization algorithms in 2D [20], as well as extensions to 3D [26, 27]. The dual of the Voronoi diagram, the Delaunay triangulation (or tetrahedralization in 3D) has also been used extensively. Here the skeleton is defined as the locus of centers of circumscribed circles of each triangle (spheres of each tetrahedra in 3D) [9, 19]. Both types of methods preserve topology and accurately localize skeletal points, provided that the boundary is sampled densely. Unfortunately, the techniques used to prune elements of the Voronoi graph which correspond to small perturbations of the boundary are typically based on heuristics. In practice, the results are not invariant under Euclidean transformations and the optimization step, particularly in 3D, can have a high computational complexity [19].

A third class of methods exploits the fact that the lo-

cus of skeletal points coincides with the singularities of a *Euclidean distance function* to the boundary. These approaches attempt to detect local maxima of the distance function, or the corresponding discontinuities in its derivatives [1, 12, 10]. The numerical detection of these singularities is itself a non-trivial problem; whereas it may be possible to localize them, ensuring homotopy with the original object is difficult.

This paper presents an overview of a novel algorithm we have developed for computing skeletons in 2D as well as in 3D which is robust and accurate, has low computational complexity, and preserves topology. The key idea is to measure the net outward flux per unit area or volume of the gradient vector field of the Euclidean distance function, and to detect locations where a conservation of energy principle is violated [28]. This is done in conjunction with a topology preserving thinning process, such that the resulting skeleton is homotopic to the original object [8, 7]. We illustrate the algorithm with several 2D and 3D examples.

2 Hamiltonian Skeletons

Consider Blum's grassfire flow [4]

$$\frac{\partial \mathcal{C}}{\partial t} = \mathcal{N} \tag{1}$$

acting on a 2D closed curve $\mathcal{C}$, such that each point on its boundary is moving with unit speed in the direction of the inward normal $\mathcal{N}$. In recent work, we have shown that this formulation leads to a Hamilton-Jacobi equation on the Euclidean distance function to the initial curve [28]. In physics, such equations are typically solved by looking at the evolution of the phase space of the equivalent Hamiltonian system. Since Hamiltonian systems are conservative, the locus of skeletal points coincides with locations where a conservation of energy principle is violated. This loss of energy can be used to formulate a natural criterion for detecting singularities of the distance function.

More specifically, let D be the Euclidean distance function to the initial surface $\mathcal{C}_0$. The magnitude of its gradient, $\| \nabla D \|$, is identical to 1 in its smooth regime. With $\mathbf{q} = (x, y)$, $\mathbf{p} = (D_x, D_y)$, the Hamiltonian system is given by

$$\dot{\mathbf{p}} = -\frac{\partial H}{\partial \mathbf{q}} = (0,0), \qquad \dot{\mathbf{q}} = \frac{\partial H}{\partial \mathbf{p}} = -(D_x, D_y), \tag{2}$$

with an associated Hamiltonian function $H = 1 - (D_x^2 + D_y^2)^{\frac{1}{2}}$. It is straightforward to show that all Hamiltonian systems are conservative [21, p. 172]; hence they are divergence free. Conversely, when trajectories of the system intersect, a conservation of energy principle is violated. This suggests a natural approach for detecting the skeleton: compute the divergence of the gradient vector field $\dot{\mathbf{q}}$ and detect locations where it is not zero. The divergence is defined as the net outward flux per unit area, as the area about the point shrinks to zero:

$$\mathrm{div}(\dot{\mathbf{q}}) \equiv \lim_{\Delta \mathrm{a} \to 0} \frac{\int_{\mathrm{L}} < \dot{\mathbf{q}}, \mathcal{N} > \mathrm{dl}}{\Delta \mathrm{a}} \tag{3}$$

Here Δa is the area, L is its bounding contour and $\mathcal{N}$ is the outward normal at each point on the contour. Via the divergence theorem

$$\int_a \mathrm{div}(\dot{\mathbf{q}}) \mathrm{da} \equiv \int_{\mathrm{L}} < \dot{\mathbf{q}}, \mathcal{N} > \mathrm{dl}. \tag{4}$$

In other words, the integral of the divergence of the vector field within a finite area gives the net outward flux through the contour which bounds it. Locations where the flux is negative, and hence energy is lost, correspond to sinks or skeletal points of the interior. Similarly, locations where the flux is positive correspond to sources or skeletal points of the exterior.[1] The extension of the above analysis to 3D is straightforward: simply replace the initial closed curve $\mathcal{C}$ with a closed surface $\mathcal{S}$ in Eq. (1), add a third coordinate z to the phase space in Eq. (2), and replace the area element with a volume element and the contour integral with a surface integral in Eq. (3).

Figure 1 illustrates the divergence-based computation on the silhouette of a panther shape. The gradient vector field of the Euclidean distance function is shown on the top, with the total outward flux on the bottom. Observe that the smooth regime of the vector field gives zero flux (medium gray), its sinks coincide with the medial axis of the interior (black), and its sources with the medial axis of the exterior (light gray). Hence, a threshold on the divergence map yields a close approximation to the skeleton, as used in [28]. However, in general it is impossible to guarantee that the result is homotopic to the original shape by simple thresholding. A high threshold may yield a connected set, but it is not thin and unwanted branches may be present, Figure 2 (top). A low threshold yields a thin set, but it may be disconnected, Figure 2 (bottom). The solution, as we shall now show, is to introduce additional constraints to ensure that the topology of the shape is preserved. The main idea is to incorporate a homotopy preserving thinning process, where the removal of points is guided by their divergence values.

3 Homotopy Preserving Skeletons

Our goal is to combine the divergence computation with a thinning process, such that as many points as possible are removed without altering the object's topology. In digital

[1] Note that the classical definition of divergence as the sum of the partial derivatives of the vector field in its component directions cannot be used where it is singular, and these are precisely the points we are interested in.

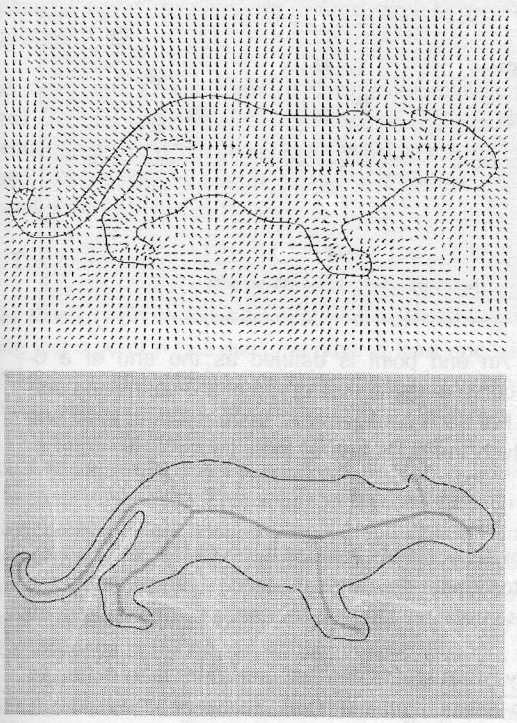

Figure 1: The gradient vector field of a signed distance function to the boundary of a panther shape (top), with the associated total outward flux (bottom). Whereas the smooth regime of the vector field gives zero flux (medium gray), its sinks correspond to the medial axis of the interior (dark gray) and its sources to the medial axis of the exterior (light gray).

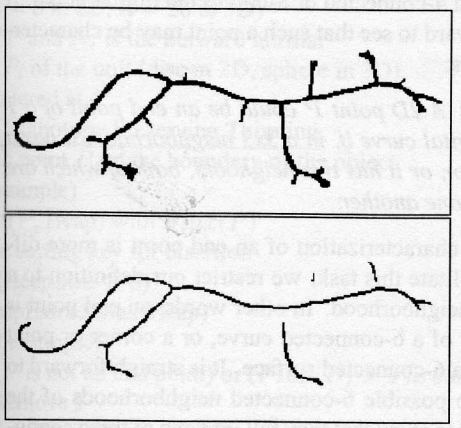

Figure 2: Thresholding the divergence map in Figure 1. A high threshold yields a connected set, but it is not thin, and unwanted branches are present (top). A low threshold yields a closer approximation to the desired medial axis, but the result is now disconnected (bottom).

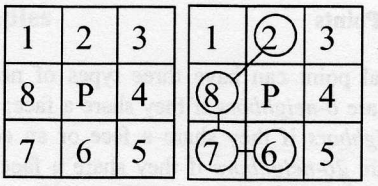

Figure 3: LEFT: A 3x3 neighborhood of a candidate point for removal P. RIGHT: An example neighborhood graph for which P is simple. Note that there is no edge between neighbors 6 and 8 (see text).

topology a point is *simple* if its removal does not change the topology of the object. In 2D we shall consider rectangular lattices, where a point is a unit square with 8 neighbors, as shown in Figure 3 (left). Hence, a 2D digital point is simple if its removal does not disconnect the object or create a hole. In 3D we shall consider cubic lattices, where a point is a unit cube with 6 faces, 12 edges and 8 vertices. Hence, a 3D digital point is simple if its removal does not disconnect the object, create a hole, or create a cavity.

3.1 2D Simple Points

Consider the 3x3 neighborhood of a 2D digital point P contained within an object, and select those neighbors which are also contained within the object. Construct a neighborhood graph by placing edges between all pairs of neighbors (not including P) that are 4-adjacent or 8-adjacent to one another. If any of the 3-tuples $\{2,3,4\}$, $\{4,5,6\}$, $\{6,7,8\}$, or $\{8,1,2\}$, are nodes of the graph, remove the corresponding diagonal edges $\{2,4\}$, $\{4,6\}$, $\{6,8\}$, or $\{8,2\}$, respectively. This ensures that there are no degenerate cycles in the neighborhood graph (cycles of length 3). Now, observe that if the removal of P disconnects the object, or introduces a hole, the neighborhood graph will not be connected, or will have a cycle, respectively. Conversely, a connected graph that has no cycles is a tree. Hence, we have a criterion to decide whether or not P is simple:

Proposition 1 *A 2D digital point P is* simple *if and only if its 3x3 neighborhood graph, with cycles of length 3 removed, is a tree.*

A straightforward way of determining whether or not a graph is a tree is to check that its Euler characteristic $|V| - |E|$ (the number of vertices minus the number of edges) is identical to 1. Note that this check only has to be performed locally, in the 3x3 neighborhood of P. Figure 3 (right) shows an example neighborhood graph for which P can be removed.

3.2 3D Simple Points

In 3D a digital point can have three types of neighbors: two points are *6-neighbors* if they share a face; two points are *18-neighbors* if they share a face or an edge; and two points are *26-neighbors* if they share a face, an edge or a vertex. This induces three *n-connectivities*, where $n \in \{6, 18, 26\}$, as well as three *n-neighborhoods* for x ($N_n(x)$). A n-neighborhood without its central point is defined as $N_n^* = N_n(x) \backslash \{x\}$. An object A is *n-adjacent* to an object B, if there exist two points $x \in A$ and $y \in B$ such that x is an n-neighbor of y. A *n-path* from x_1 to x_k is a sequence of points $x_1, x_2, ..., x_k$, such that for all x_i, $1 < i \leq k$, x_{i-1} is n-adjacent to x_i. An object represented by a set of points O is *n-connected*, if for every pair of points $(x_i, x_j) \in O \times O$, there is a n-path from x_i to x_j.

Based on these definitions, Malandain et al. provide a topological classification of a point x in a cubic lattice by computing two numbers [16]: i) C^*: the number of 26-connected components 26-adjacent to x in $O \cap N_{26}^*$, and ii) $\bar{C}$: the number of 6-connected components 6-adjacent to x in $\bar{O} \cap N_{18}$. An important result with respect to our goal of thinning is stated in the following theorem:

Theorem 1 *(Malandain et al. 1993) P is* simple *if $C^*(P) = 1$ and $\bar{C}(P) = 1$.*

We can now determine whether or not the removal of a point will alter the topology of a digital object. When ensuring homotopy is the only concern, simple points can be removed sequentially until no more simple points are left. The resulting set will be thin and homotopic to the object. However, the relationship to the skeleton will be uncertain since the locus of surviving points depends entirely on the order in which the simple points are removed. In the current context, we have derived a natural criterion for ordering the thinning, based on the divergence of the gradient vector field of the Euclidean distance function.

3.3 Divergence-Ordered Thinning

Recall from Section 2, that a conservation of energy principle is violated at skeletal points. The total outward flux of the gradient vector field of the Euclidean distance function is negative at such points, since they correspond to sinks. [2] The magnitude of the total outward flux is proportional to the amount of energy absorbed, and hence provides a natural measure of the "strength" of a skeletal point for numerical computations. The essential idea is to order the thinning such that the weakest points are removed first, and to stop the process when all surviving points are not simple, or have a total outward flux below some chosen (negative) value, or both. This will accurately localize the skeleton, and also ensure homotopy with the original object. Unfortunately the result is not guaranteed to be a thin set, i.e., one without an interior.

[2] Conversely, skeletal points of the background correspond to sources, with positive total outward flux.

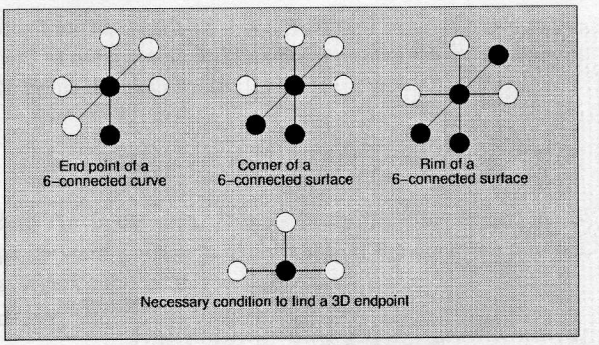

Figure 4: An end point is defined as the end of a 6-connected curve or the corner or rim of a 6-connected surface in 3D. For each configuration, there exists at least one plane in which the point has at least three background 6-neighbors.

One way of satisfying this last constraint is to define an appropriate notion of an end point. Such a point would correspond to the end point of a curve (in 2D or 3D), or a point on the rim of a surface, in 3D. The thinning process would proceed as before, but the threshold criterion for removal would be applied *only to end points*. Hence, all surviving points which were not end points would not be simple, and the result would be a thin set.

In 2D, an end point will be viewed as any point that could be the end of a 4-connected or 8-connected digital curve. It is straightforward to see that such a point may be characterized as follows:

Proposition 2 *A 2D point P could be an end point of a 1 pixel thick digital curve if, in a 3x3 neighborhood, it has a single neighbor, or it has two neighbors, both of which are 4-adjacent to one another.*

In 3D, the characterization of an end point is more difficult. To facilitate this task, we restrict our definition to a 6-connected neighborhood. In other words, an end point is either the end of a 6-connected curve, or a corner or point on the rim of a 6-connected surface. It is straightforward to enumerate the possible 6-connected neighborhoods of the end point, and to show that they fall into one of three configurations (see Figure 4). Notice that for each configuration, there exists at least one plane in which the point has at least three background 6-neighbors, and that the end point must also be a simple point:

Proposition 3 *A 3D point P could be an end point of a 6-connected curve, or a corner or point on the rim of a 6-connected surface, if it is simple and there exists at least*

one plane in which the point has at least three background 6-neighbors.

We now have all the tools to design our divergence-ordered thinning process. Simple points will removed sequentially, ordered by divergence, until a threshold is reached. Subsequently, simple points will continue to be removed if they are not end points.

3.4 The Algorithm

The thinning process can be made very efficient by observing that a point which does not have at least one background point as an immediate neighbor cannot be removed, since this would create a hole or a cavity. Therefore, the only potentially removable points are on the border of the object. Once a border point is removed, only its neighbors may become removable. This suggests the implementation of the thinning process using a heap. A full description of the procedure is given in Algorithm 1; its worst-case complexity can be shown to be $\mathcal{O}(n) + \mathcal{O}(klog(k))$, where n is the total number of points in the 2D or 3D array and k is the number of points in the interior of the object [7].

Algorithm 1 The divergence-ordered thinning algorithm.

Part I: Total Outward Flux
Compute the distance transform of the object D [5].
Compute the gradient vector field ∇D.
Compute the net outward flux of ∇D using Eq. 4
  For each point P in the interior of the object
    $Flux(P) = \sum_{i=1}^{n} < N_i, \nabla D(P_i) >$,
    where P_i is an n-neighbor
    ($n = 8$ in 2D, $n = 26$ in 3D)
    of P and N_i is the outward normal
    at P_i of the unit (disc in 2D, sphere in 3D),
    centered at P.
Part II: Homotopy Preserving Thinning
For each point P on the boundary of the object
if (P is simple)
  insert(P, Heap) with $Flux(P)$
  as the sorting key for insertion
While (Heap.size > 0)
P = HeapExtractMax(Heap)
  if (P is simple)
    if (P is not an end point) or ($Flux(P) > Thresh$)
      Remove P
      for all neighbors Q of P
        if (Q is simple)
          insert(Q, Heap)
    else mark P as a skeletal (end) point
    end { if }
  end { if }
end { while }

4 Examples

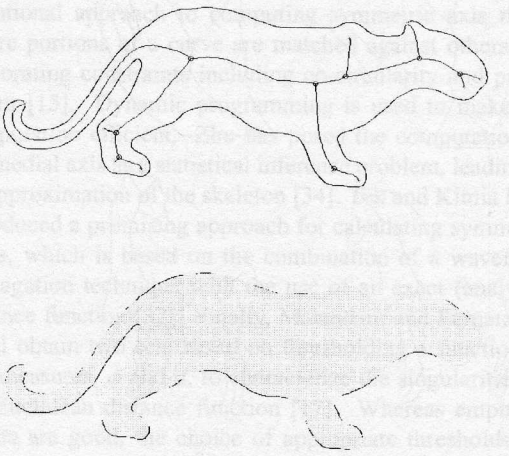

Figure 5: TOP: A subpixel medial axis, with branch points shown as empty circles and end points as filled circles. Compare with the results in Figure 2. BOTTOM: The reconstruction as the envelope of the maximal inscribed disks (grey) of the medial axis, overlayed on the original shape.

We first present examples of medial axes, computed for a range of shapes. The same divergence threshold was used in each example, to determine which end points to preserve. The implementation uses an exact distance function to a piecewise circular arc interpolation of the boundary, which allows for subpixel computations (details are presented in [8]). Figure 5 (top) shows the subpixel medial axis for the panther silhouette with branch points shown as empty circles and end points as closed circles. The accuracy of the representation is illustrated in Figure 5 (bottom), where the shape is reconstructed as the envelope of the maximal inscribed discs associated with each medial axis point. Figure 6 depicts subpixel medial axes for a number of other shapes. The results demonstrate the robustness of the framework under Euclidean transformations, as well as changes in scale.

Next, we show medial surfaces computed for synthetic data as well as volumes segmented from MR and MRA images. For these we used the D-Euclidean distance function [5] which provides a good approximation to the true distance function. Once again, the only free parameter is the choice of the divergence threshold below which the removal of end points is blocked. For these examples, the value was selected so that approximately 25% of the points within the volume had a lower divergence.

Figure 7 compares our approach with the parallel thinning method introduced by Manzanera et al. [18]. Note that

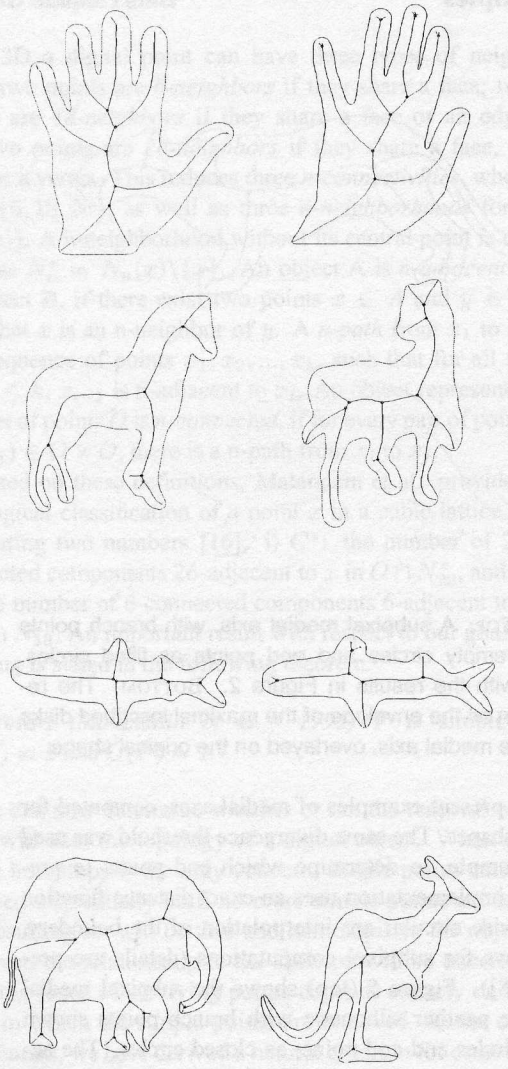

Figure 6: Subpixel medial axes for a range of shapes, obtained by divergence-ordered thinning.

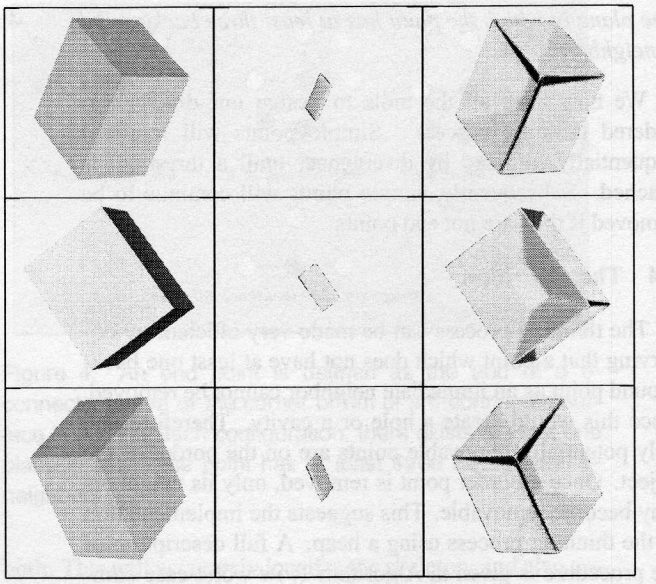

Figure 7: LEFT COLUMN: Three views of a cube. MIDDLE COLUMN: The corresponding medial surfaces computed using the algorithm of [18]. RIGHT COLUMN: The corresponding divergence-based medial surfaces.

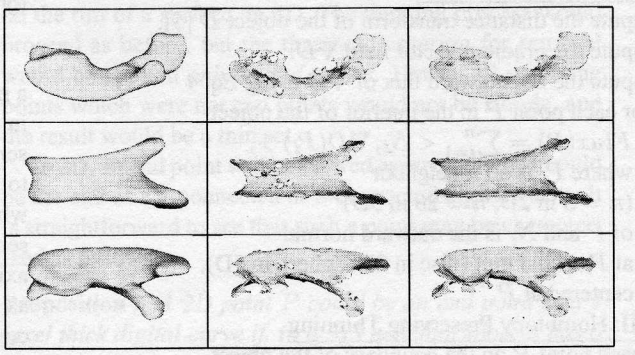

Figure 8: LEFT COLUMN: Three views of the ventricles of a brain, segmented from volumetric MR data using an active surface. MIDDLE COLUMN: The corresponding medial surfaces obtained by thresholding the divergence. RIGHT COLUMN: The divergence-based medial surfaces obtained using the same threshold, but with the incorporation of homotopy preserving thinning.

the latter method yields only a subset of the "true" medial surface for these data sets, and hence only a coarse approximation to the object is possible. In contrast, the divergence-based medial surfaces carry more information, from which a near perfect reconstruction is possible.

Recall from Section 2 that thresholding the total outward flux provides an approximation to the skeleton at low computational cost. However, as we showed in the 2D case (Figure 2), if the threshold is too high, slight perturbations of the boundary will be represented, and the resulting structure will not be a thin set. On the other hand, lowering the threshold can provide a thin set, but at the cost of altering the object's topology. This is illustrated for a 3D object in Figure 8 (middle column) where the medial surfaces corresponding to the views in the left column are accurate and thin, but have holes. Observe that with the same threshold as before, the divergence-based thinning algorithm now yields a thin structure which preserves topology (Figure 8 (right column)).

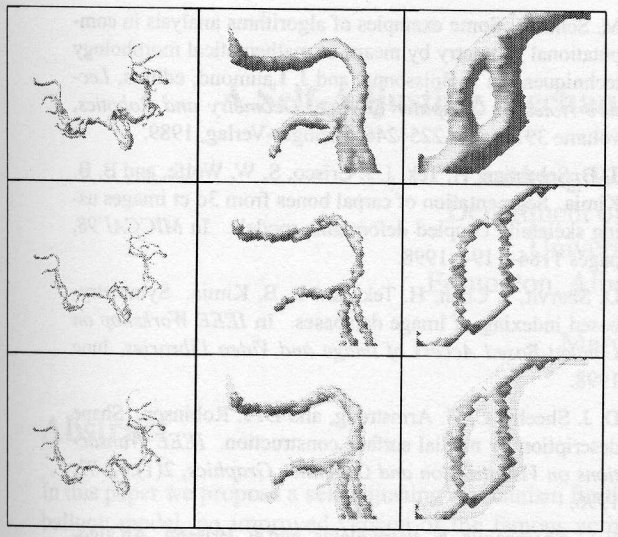

Figure 9: TOP ROW: Blood vessels segmented from volumetric MRA data, with details of parts shown in the middle and right columns. MIDDLE ROW: The corresponding divergence-based medial surfaces. THIRD ROW: The divergence-based medial surfaces (solid) are shown within the vessel surfaces (transparent).

Finally, we illustrate the robustness of the approach on a data set of blood vessels obtained from an MRA image of the brain, in Figure 9. The blood vessels have complex topology with loops, and are already quite thin in several places. The bottom row illustrates the accuracy of the method, where the medial surfaces are shown embedded within the original data set. Generically these structures are thin sheets which approach 3D curves when the blood vessels become perfectly cylindrical.

5 Conclusions

We have presented an overview of a novel algorithm for computing 2D and 3D skeletons which is robust and accurate, computationally efficient, invariant to Euclidean transformations and homotopy preserving. The essential idea is to combine a divergence computation on the gradient vector field of the Euclidean distance function to the object's boundary with a thinning process that preserves topology. In 2D, the digital medial axis is also shifted to be within arbitrary precision of the "true" medial axis, and may be directly interpreted as a skeletal graph [8]. In 3D, the characterization of simple (or removable) points is adopted from [16], but we have also introduced the notion of an end point of a 6-connected structure, in order that the algorithm may converge to a thin set [7]. We have illustrated the advantages of the approach on synthetic and real binary volumes of varying complexity.

We note that in related work, Leymarie and Levine utilized the magnitude of the gradient vector field to design a potential function to attract a snake moving in from the object's boundary [12]. Geiger et. al have introduced a variational approach to computing symmetric axis trees, where portions of a curve are matched against others, incorporating constraints including co-circularity and parallelism [15]. Dynamic programming is used to make the computation efficient. Zhu has posed the computation of the medial axis as a statistical inference problem, leading to an approximation of the skeleton [34]. Tek and Kimia have introduced a promising approach for calculating symmetry maps, which is based on the combination of a wavefront propagation technique with the use of an exact (analytic) distance function [32]. Finally, Malandain and Fernandez-Vidal obtain two sets based on thresholding a function of two measures, ϕ and d, to characterize the singularities of the Euclidean distance function [17]. Whereas empirical results are good, the choice of appropriate thresholds for these measures, as well as strategies for combining them, are based on heuristics.

In contrast, our method is rooted in a physics-based analysis of the gradient vector field of the Euclidean distance function, which shows that a conservation of energy principle is violated at skeletal points and provides a single consistent framework within which to compute skeletons in 2D and 3D. We use the divergence theorem to compute the total outward flux of the vector field, and to locate points where energy is absorbed. It should be clear that whereas we have focussed on the interior of an object, the skeleton of the background can be similarly obtained by locating points that act as sources, and have positive total outward flux.

Acknowledgements We are grateful to Allen Tannenbaum and Steve Zucker for collaborations on the hamilton-jacobi formulation. Louis Collins, Georges Le Goualher, Belinda Lee and Terry Peters kindly supplied the medical data. This research was supported by CFI, FCAR and NSERC.

References

[1] C. Arcelli and G. S. di Baja. Ridge points in euclidean distance maps. *Pattern Recognition Letters*, 13(4):237–243, 1992.

[2] C. Arcelli and G. Sanniti di Baja. A width-independent fast thinning algorithm. *IEEE PAMI*, 7(4):463–474, July 1985.

[3] G. Bertrand. A parallel thinning algorithm for medial surfaces. *Pattern Recognition Letters*, 16:979–986, 1995.

[4] H. Blum. Biological shape and visual science. *Journal of Theoretical Biology*, 38:205–287, 1973.

[5] G. Borgefors. Distance transformations in arbitrary dimensions. *CVGIP*, 27:321–345, 1984.

[6] G. Borgefors, I. Nystrom, and G. S. D. Baja. Computing skeletons in three dimensions. *Pattern Recognition*, 32:1225–1236, 1999.

[7] S. Bouix and K. Siddiqi. Divergence-based medial surfaces. In *ECCV'2000*, Dublin, Ireland, June 2000.

[8] P. Dimitrov, C. Phillips, and K. Siddiqi. Robust and efficient skeletal graphs. In *CVPR'2000*, Hilton Head, South Carolina, June 2000.

[9] J. A. Goldak, X. Yu, A. Knight, and L. Dong. Constructing discrete medial axis of 3-d objects. *Int. Journal of Computational Geometry and Applications*, 1(3):327–339, 1991.

[10] J. Gomez and O. Faugeras. Reconciling distance functions and level sets. Technical Report TR3666, INRIA, April 1999.

[11] T.-C. Lee and R. L. Kashyap. Building skeleton models via 3-d medial surface/axis thinning algorithm. *CVGIP: Graphical Models and Image Processing*, 56(6):462–478, November 1994.

[12] F. Leymarie and M. D. Levine. Simulating the grassfire transform using an active contour model. *IEEE PAMI*, 14(1):56–75, Jan. 1992.

[13] A. Liu, E. Bullitt, and S. M. Pizer. 3d/2d registration via skeletal near projective invariance in tubular objects. In *MICCAI'98*, pages 952–963, 1998.

[14] T.-L. Liu and D. Geiger. Approximate tree matching and shape similarity. In *ICCV'99*, pages 456–461, Kerkyra, Greece, September 1999.

[15] T.-L. Liu, D. Geiger, and R. V. Kohn. Representation and self-similarity of shapes. In *Sixth International Conference on Computer Vision*, 1998.

[16] G. Malandain, G. Bertrand, and N. Ayache. Topological segmentation of discrete surfaces. *Int. Journal of Computer Vision*, 10(2):183–197, 1993.

[17] G. Malandain and S. Fernandez-Vidal. Euclidean skeletons. *Image and Vision Computing*, 16:317–327, 1998.

[18] A. Manzanera, T. M. Bernard, F. Preteux, and B. Longuet. Medial faces from a concise 3d thinning algorithm. In *ICCV'99*, pages 337–343, Kerkyra, Greece, September 1999.

[19] M. Näf, O. Kübler, R. Kikinis, M. E. Shenton, and G. Székely. Characterization and recognition of 3d organ shape in medical image analysis using skeletonization. In *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, 1996.

[20] R. L. Ogniewicz. *Discrete Voronoi Skeletons*. Hartung-Gorre, 1993.

[21] L. Perko. *Differential Equations and Dynamical Systems*. Springer-Verlag, 1986.

[22] S. M. Pizer, A. Thall, and D. T. Chen. M-reps: A new object representation for graphics. *ACM Transactions on Graphics*, submitted, 1999.

[23] M. Schmitt. Some examples of algorithms analysis in computational geometry by means of mathematical morphology techniques. In J. Boissonnat and J. Laumond, editors, *Lecture Notes in Computer Science, Geometry and Robotics*, volume 391, pages 225–246. Springer-Verlag, 1989.

[24] T. B. Sebastian, H. Tek, J. J. Crisco, S. W. Wolfe, and B. B. Kimia. Segmentation of carpal bones from 3d ct images using skeletally coupled deformable models. In *MICCAI'98*, pages 1184–1194, 1998.

[25] D. Sharvit, J. Chan, H. Tek, and B. B. Kimia. Symmetry-based indexing of image databases. In *IEEE Workshop on Content-Based Access of Image and Video Libraries*, June 1998.

[26] D. J. Sheehy, C. G. Armstrong, and D. J. Robinson. Shape description by medial surface construction. *IEEE Transactions on Visualization and Computer Graphics*, 2(1):62–72, 1996.

[27] E. C. Sherbrooke, N. Patrikalakis, and E. Brisson. An algorithm for the medial axis transform of 3d polyhedral solids. *IEEE Transactions on Visualization and Computer Graphics*, 2(1):44–61, 1996.

[28] K. Siddiqi, S. Bouix, A. Tannenbaum, and S. W. Zucker. The hamilton-jacobi skeleton. In *ICCV'99*, pages 828–834, Kerkyra, Greece, September 1999.

[29] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, and S. W. Zucker. Shock graphs and shape matching. *International Journal of Computer Vision*, 35(1):13–32, 1999.

[30] G. D. Stetten and S. M. Pizer. Automated identification and measurement of objects via populations of medial primitives, with application to real time 3d echocardiography. In *IPMI'99*, pages 84–97, 1999.

[31] M. Teichmann and S. Teller. Assisted articulation of closed polygonal models. In *9th Eurographics Workshop on Animation and Simulation*, 1998.

[32] H. Tek and B. B. Kimia. Symmetry maps of free-form curve segments via wave propagation. In *ICCV'99*, pages 362–369, Kerkyra, Greece, September 1999.

[33] S. Zhu and A. L. Yuille. Forms: a flexible object recognition and modeling system. *International Journal of Computer Vision*, 20(3):187–212, 1996.

[34] S. C. Zhu. Stochastic computation of the medial axis in markov random fields. In *CVPR'98*, pages 72–79, Santa Barbara, CA, June 1998.