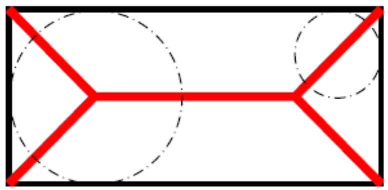
Skeleton Extrac-tion

content

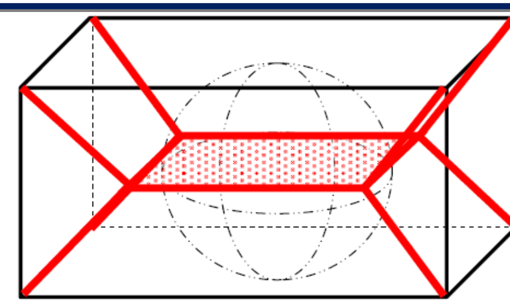
- Medial Axis
- Curve Skeletons
- Competing Front Approaches
- Mean Curvature Skeletons

MEDIAL AXIS AND CURVE SKELETONS

Medial Axis



2D Shape (black) and Medial Axis (red) grass-fire analogy



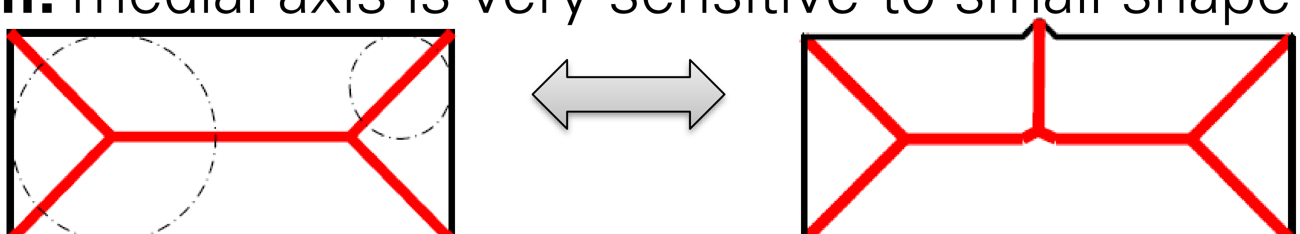
3D Shape (black) and Medial Axis (red) [only one of the 13 patches (one for each edge plus shaded) is shaded]

Definition: medial axis of a shape is the set of interior points with at least two closest points on the shape boundary.

- 2D: set of isolated points, and curve segments
- 3D: set of isolated points, curve segments and surface patches

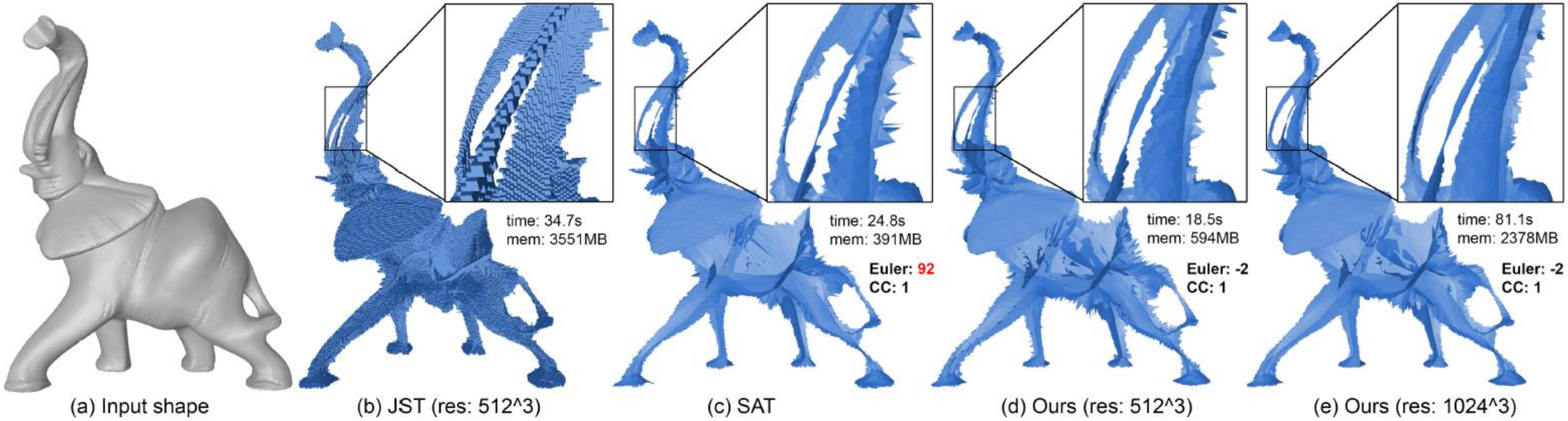
Theorem: shape can be reconstructed from medial axis attributed with radius, i.e. closest distance to shape. (medial axis transform maps shape to medial axis & radius)

Observation: medial axis is very sensitive to small shape changes



Media Axis from Voxel Cores

- Yan, Yajie, David Letscher, and Tao Ju. "Voxel Cores: Efficient, robust, and provably good approximation of 3D medial axes." *ACM Transactions on Graphics (TOG)* 37.4 (2018): 1-13. https://yajieyan.github.io/project/voxelcore-readme



- Voxel-based medial axis computation with simple algorithm and theoretical convergence strategies
  (with increasing voxel resolution convergence to true medial axis)

Media Axis from Voxel Cores

Algorithm

- Step 1: Voxelization
  voxelize input shape at user-specified voxel size h.
  (voxel belongs to the voxelization if its center is inside of input shape)
- Step 2: Extracting voxel core C
  Given a voxel shape O, compute Voronoi diagram of boundary vertices P, and keep only those Voronoi elements whose vertices lie in O.
- Step 3: λ pruning
  $\forall e \in C$, compute radius of smallest circumscribing sphere of e's nearest points in P. Given a user-specified λ, remove elements in C with radius lower than λ while maintaining the topology of C.

Voxel core Definition

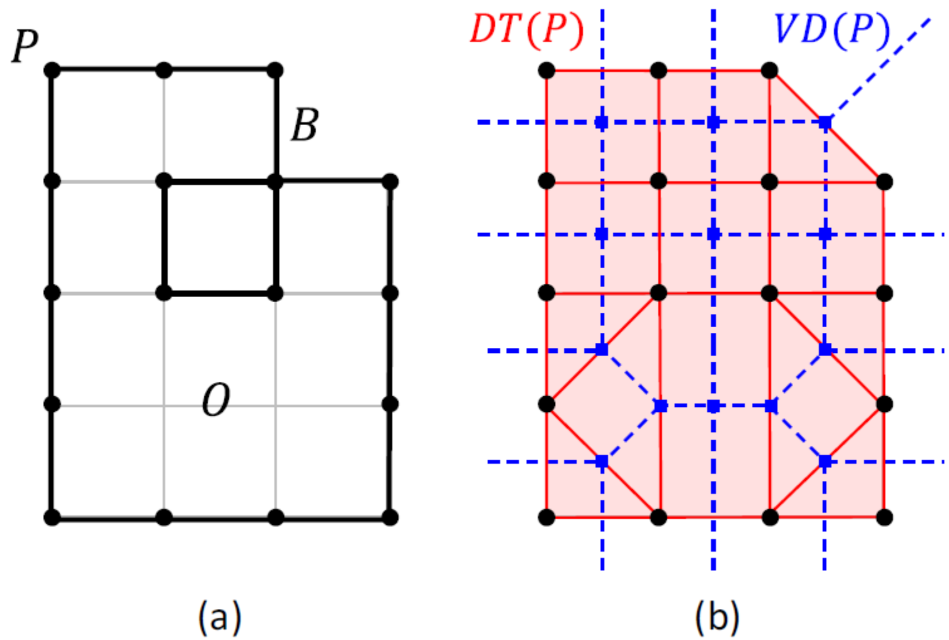


Fig. 2. Illustration of voxel core in 2D. (a) A voxel shape O with boundary set B (thick outline) and boundary vertices P (dots). (b) The Delaunay triangulation $DT(P)$ (red edges and pink cells) and Voronoi diagram $VD(P)$ (blue). (c) Subset of $DT(P)$ intersecting O and their dual Voronoi elements, which make up the voxel core C (blue).

Media Axis from Voxel Cores

Illustration of Convergence

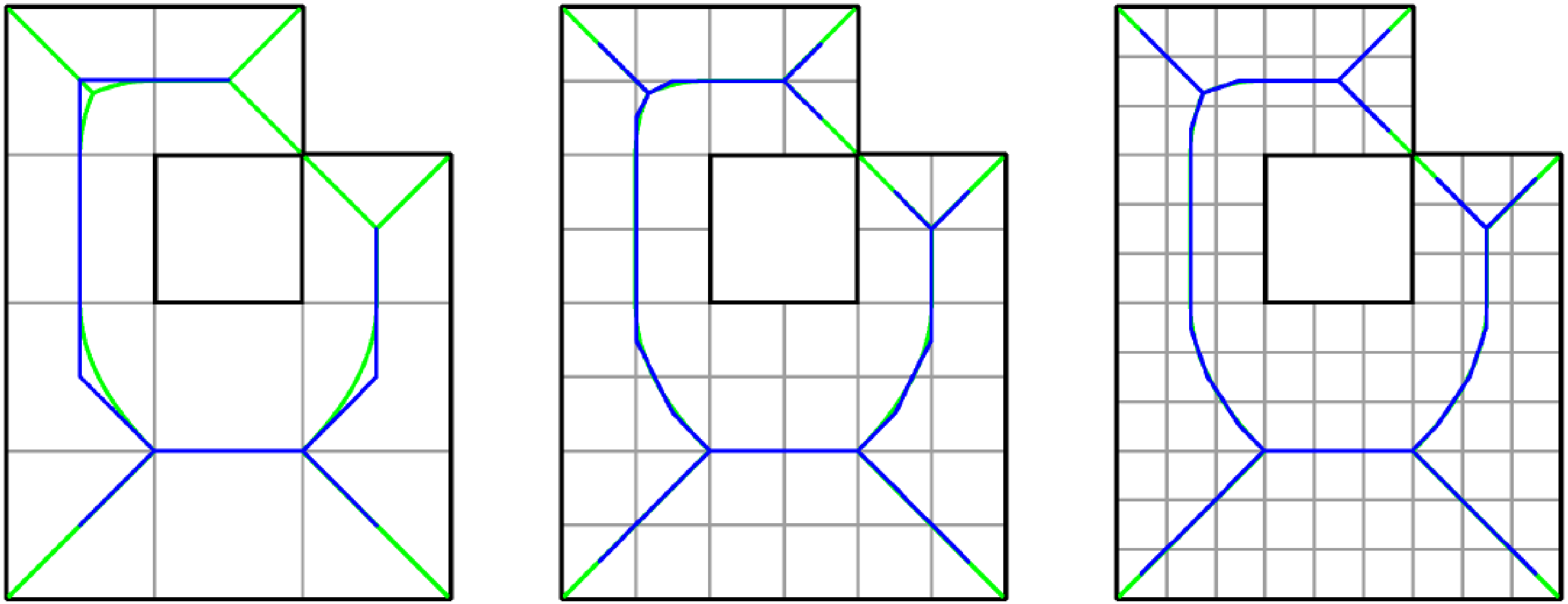
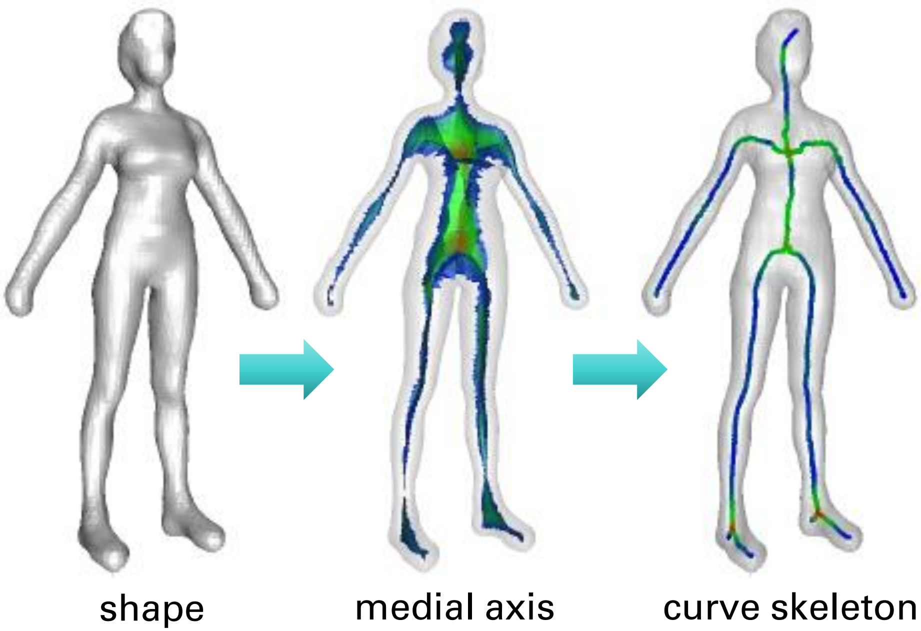
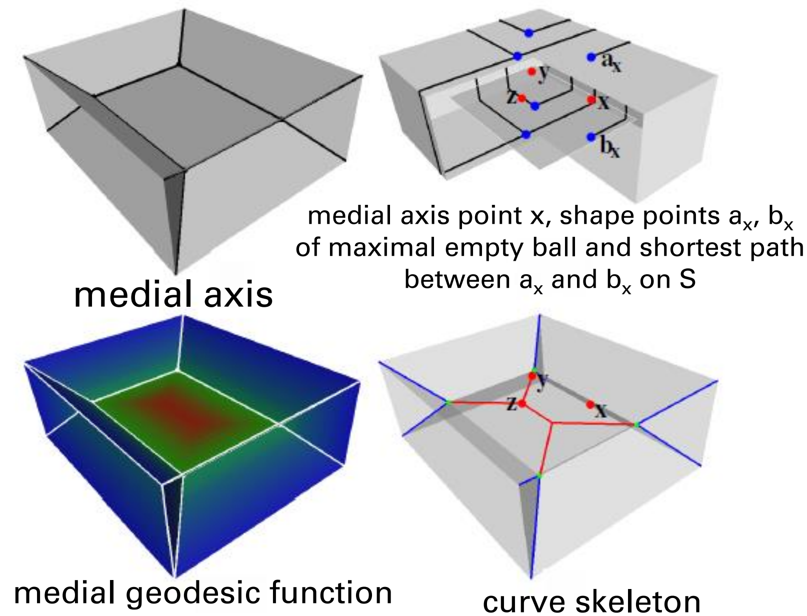


Fig. 3. Voxel core in 2D (blue) after increasing levels of voxel subdivision. Observe that it converges to the medial axis (green) of the voxel shape.

CURVE SKELETONS

Curve Skeletons



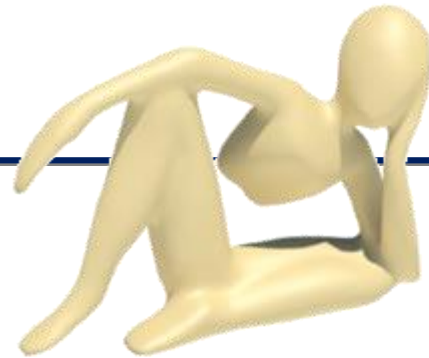


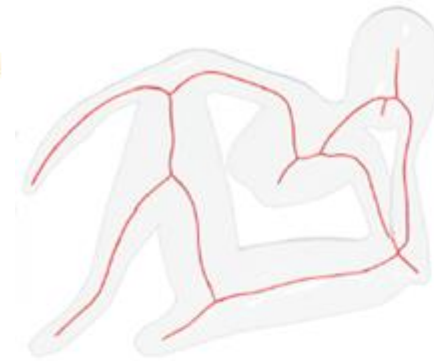
Dey, Tamal K., and Jian Sun. "Defining and computing curve-skeletons with medial geodesic function." *Symposium on geometry processing*. Vol. 6. 2006.

Three Step Definition from Dey SGP 2006

- map each point x on medial axis surface patch to shape boundary points a_x and b_x where maximal empty ball touches
- compute medial geodesic function (MGF) as geodesic distance between a_x and b_x
- define curve skeleton as curve parts of medial axis plus points where MGF is not differentiable (also curve structure)

More Curve Skeletons



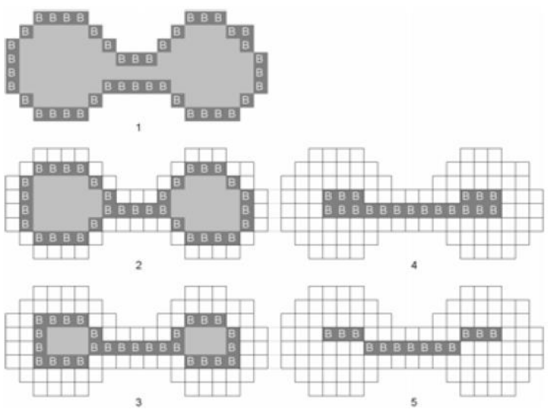


- Several more approaches that do not allow general definition
- Cornea et al. (2007) describe important properties of curve skeletons (SC):
  - thin … CS should be 1D also for 3D shapes
  - topology: CS should have same topology in a relaxed sense, i.e. same number of connected components and at least one loop per tunnel and cavity; preserve connectedness
  - invariance: CS should be invariant under isometric transformations that preserve geodesic distances between points on the surface
  - reconstruction: shape should be reconstructible from CS with minimal additional information
  - reliability: every shape point is visible from at least one skeleton point
  - centered: CS should be centered within object
  - detection of junctions and parts
  - robustness to small perturbations and curve smoothness

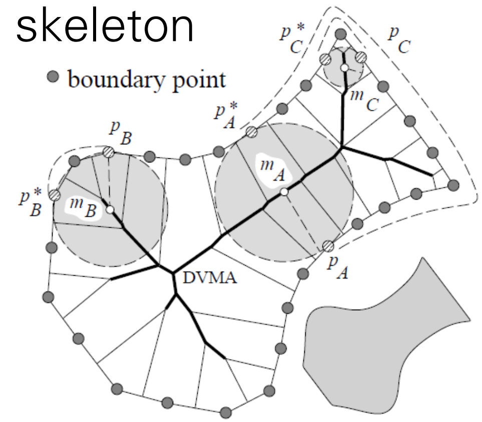
Curve Skeleton Algorithms

Computer Graphics and Visualization

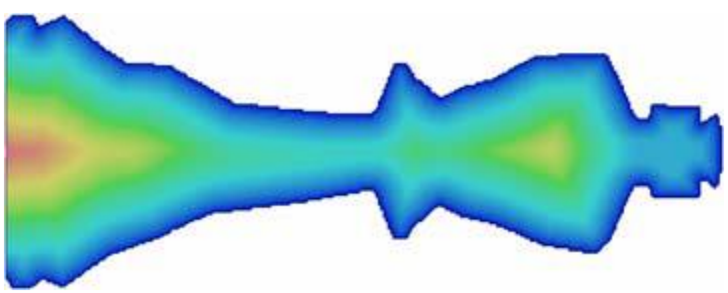
- Thinning ... discretize on volumetric grid and incrementally prune simple voxels that don't change topology



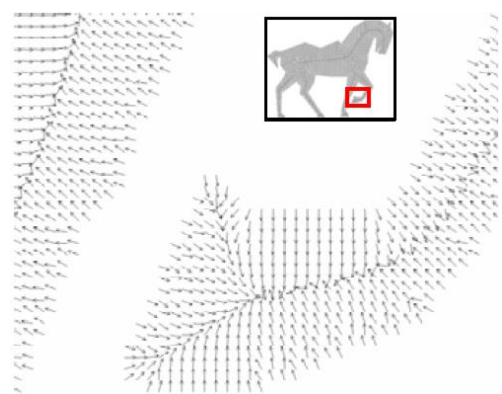
- Geometric … compute voronoi diagram and use it to discretely estimate medial axis or directly curve skeleton



- Distance field based … discretize distance to shape boundary on voxel grid and extract ridge lines (maxima along gradient direction)



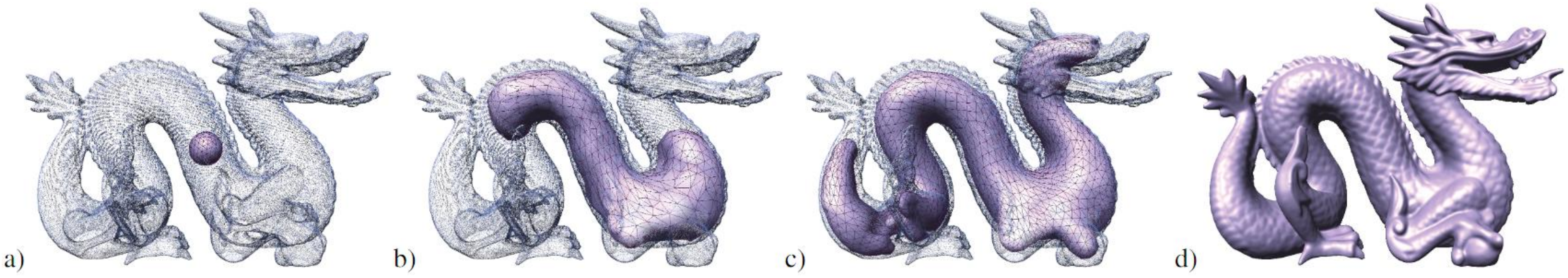
- General field functions …for example force fields and detection of local extrema



COMPETING FRONT APPROACHES

Competing Fronts

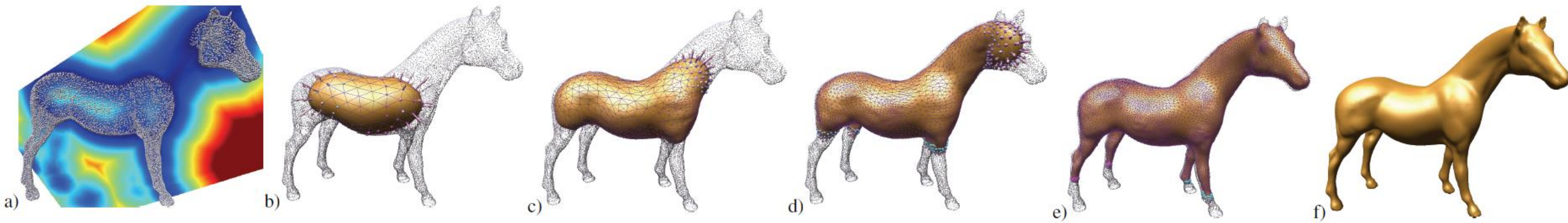
Sharf, Andrei et al. "Competing Fronts for Coarse-to-Fine Surface Reconstruction." EG 2006



- reconstruct surface by inflating deformable mesh from interior (a-c)
- do controllable topological operations (d)

Competing Fronts

Sharf, Andrei et al. "Competing Fronts for Coarse-to-Fine Surface Reconstruction." EG 2006

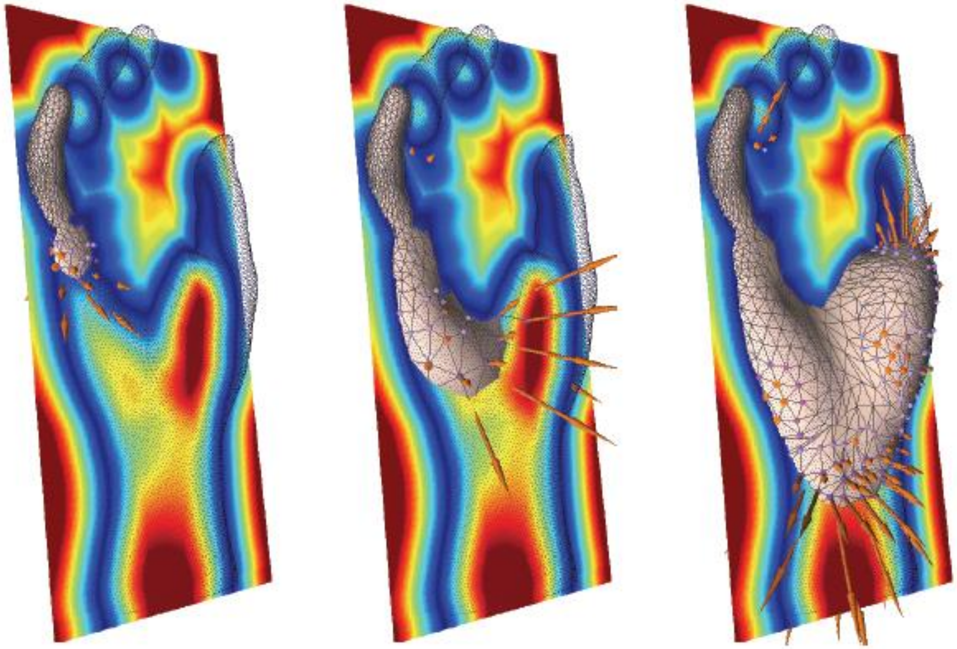


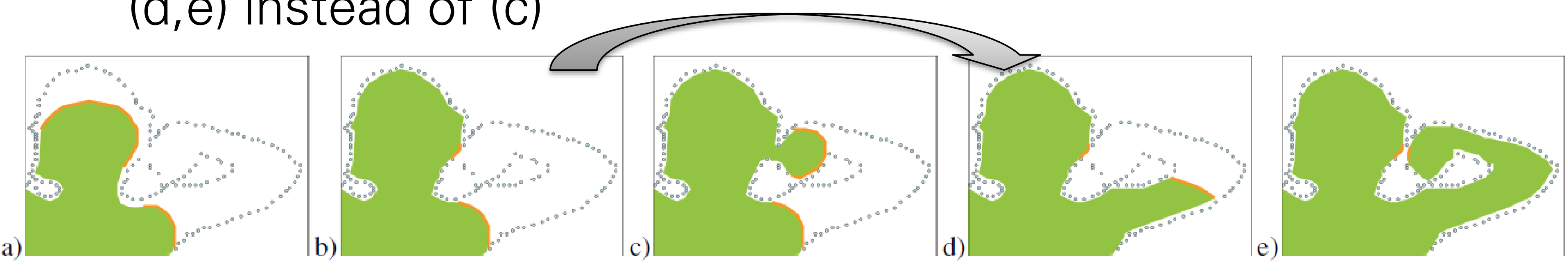
- compute unsigned distance field from points (a)
- initialize model to small sphere at maximum distance
- grow in normal direction with distance controlled step size
- keep deformable mesh smooth through Laplace operator
- tension trades smoothness vs inflation
- decrease tension and refine mesh if necessary
- project final mesh to MLS surface (f)

Competing Fronts

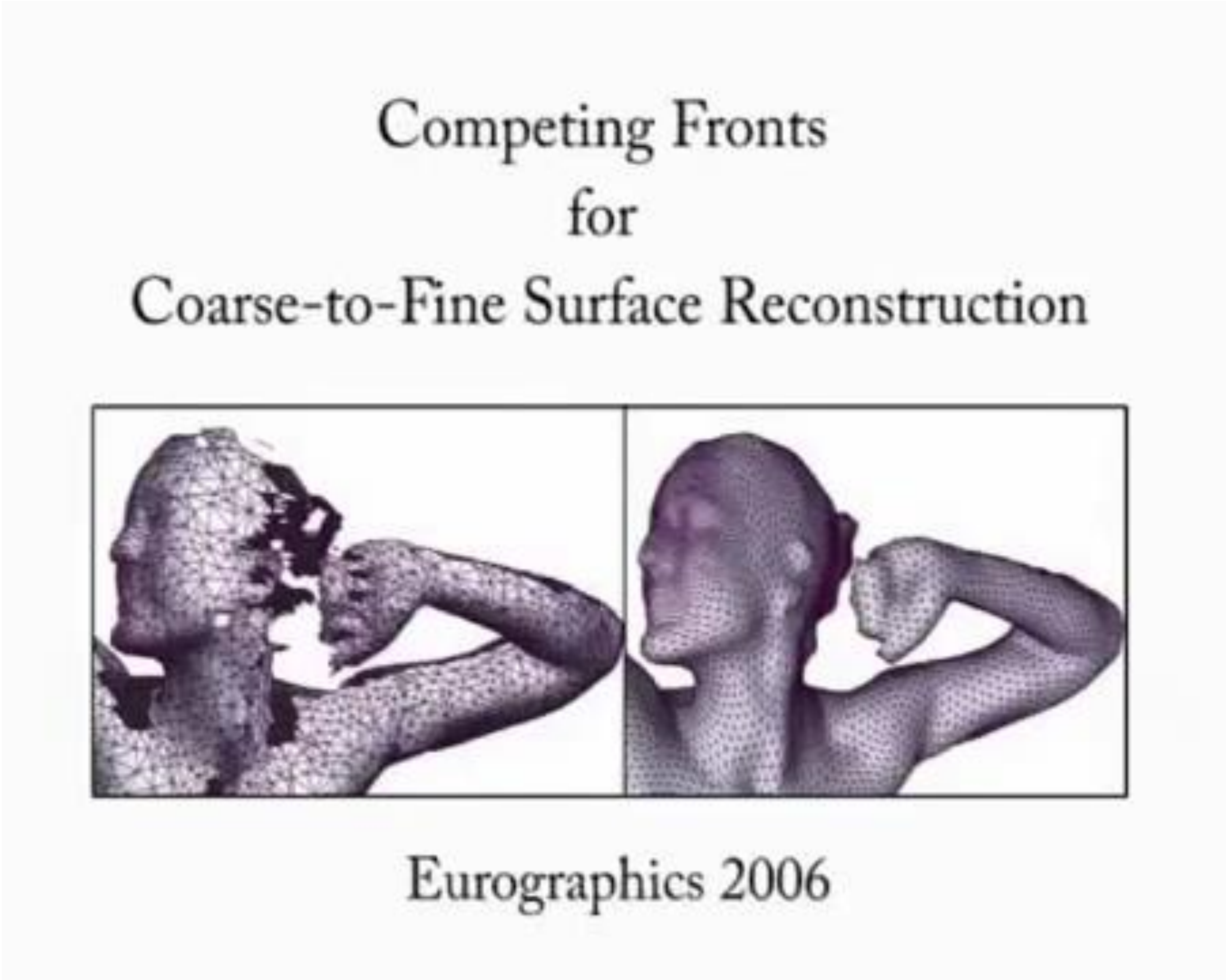
Sharf, Andrei et al. *"Competing Fronts for Coarse-to-Fine Surface Reconstruction." EG 2006*

- **right:** growth in normal direction allows passage through narrow channel even though mesh needs to grow away from surface
- restart fronts close to unreached input points
- **bottom:** coarse to fine approach avoids leaking (d,e) instead of (c)





Competing fronts for coarse-to-fine surface reconstruction



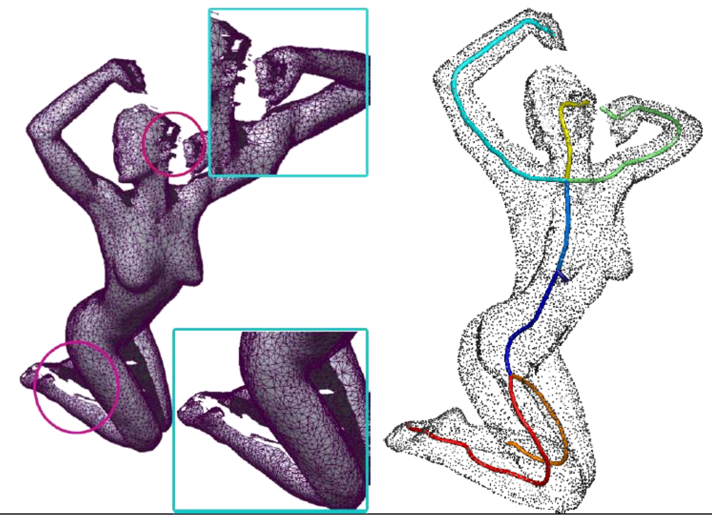
https://www.youtube.com/watch?v=IfNMZ5N2ZxQ

Fast Skeleton Extraction on Noisy Data

Sharf, Andrei, et al. "On-the-fly Curve-skeleton Computation for 3D Shapes." EG 2007.

- trace skeleton as center of competing fronts in different resolutions that evolve to reconstruct the shape
- keep book of surface to skeleton correspondences and front tension
- filter short branches of skeleton
- suitable for incomplete models and point sets

Figure 2: *The deformable model with competing fronts. The fronts move in a coarse to fine manner, and may split to form sub-fronts, inducing the branching structure of our skeleton.*



Fast Skeleton Extraction on Noisy Data

- skeleton is tracked during model growth with one branch per front
- place skeleton points in barycenter of front vertices
- track correspondence from skeleton points to mesh vertices and project to shape

postprocessing

- tension parameter is used to filter and simplify skeleton
- skeleton and tension parameter together allow for segmentation of shape into meaningful parts

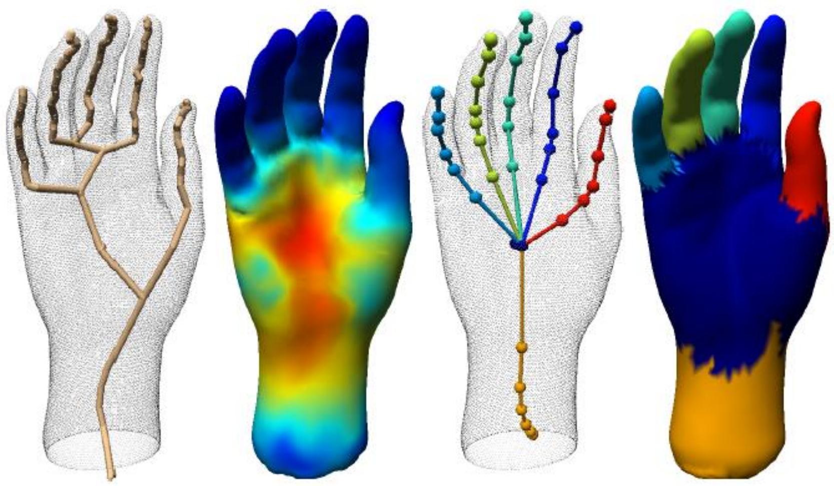


Figure 4: *Color mapping of the evolution tension parameter (left). The initial skeleton structure (left) is filtered using the evolution tension parameter (center left), simplifying the skeleton (center right) while preserving the skeleton/model correspondence and segmentation (right).*

Fast Skeleton Extraction on Noisy Data

A. Sharf & T. Lewiner & A. Shamir & L. Kobbelt / On-the-fly Curve-skeletons

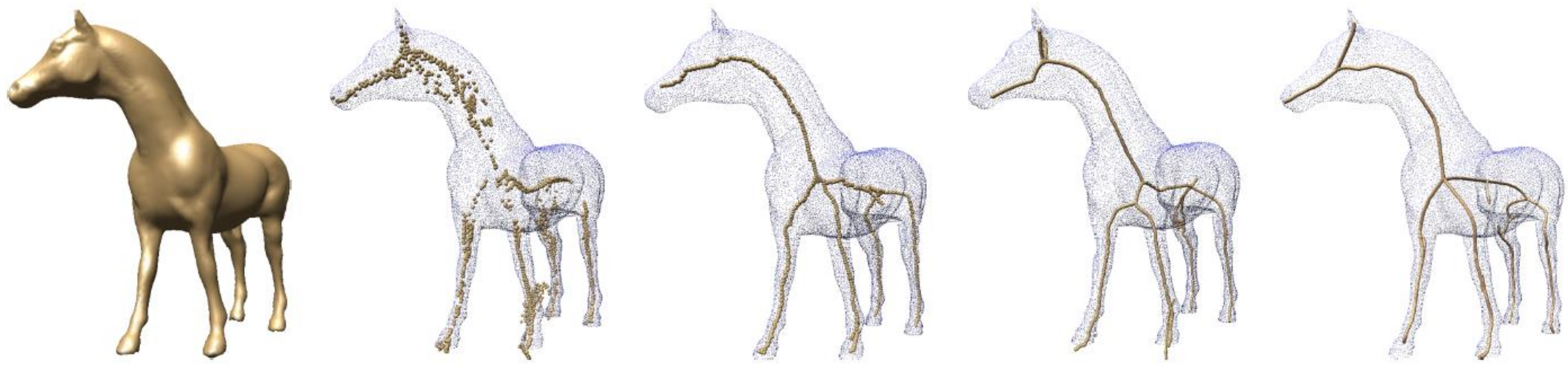
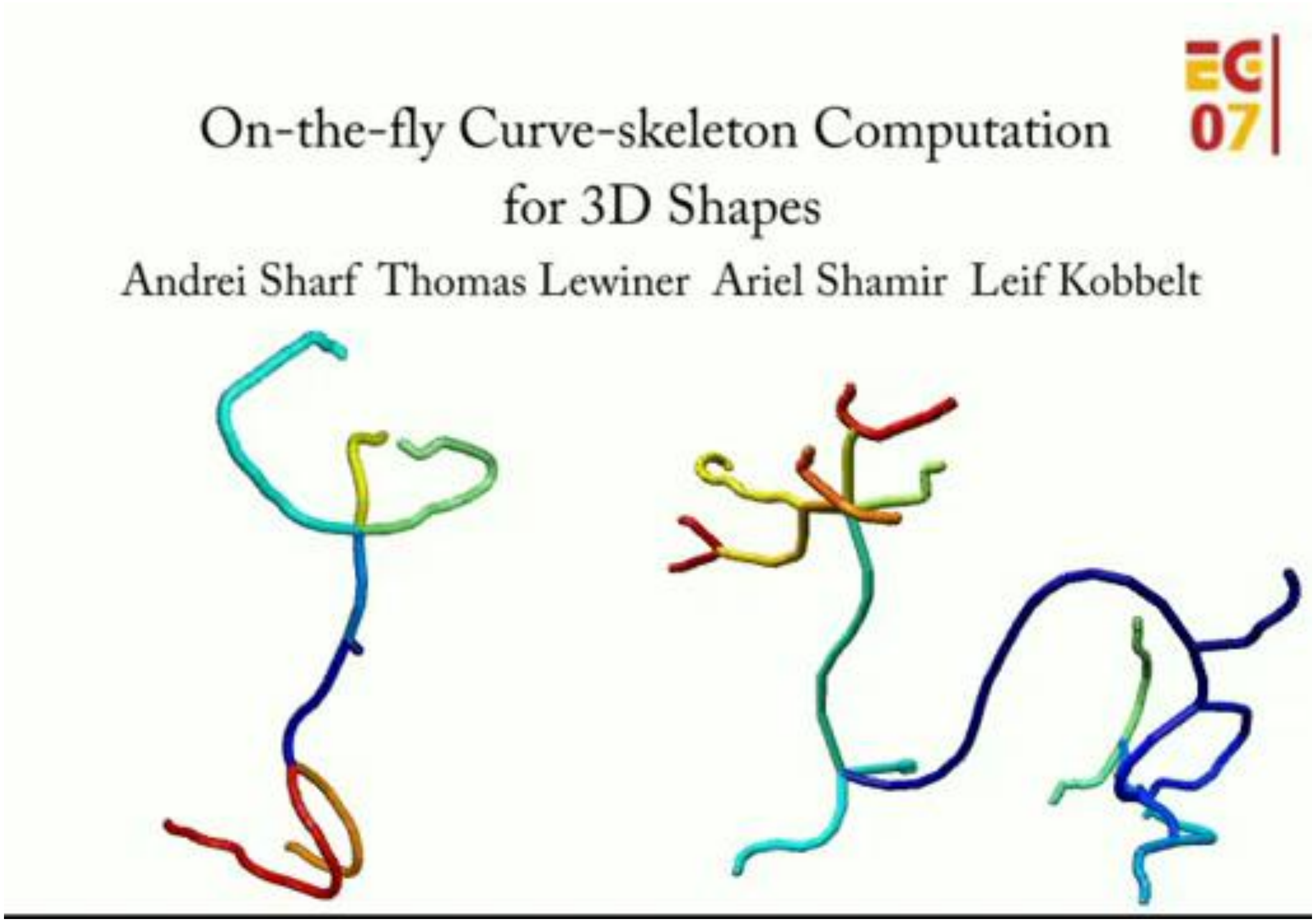


Figure 7: *Comparing our technique on a horse model. From left to right: The horse mesh. The curve-skeleton computed using the distance field of [GS99] (16 seconds). Using the topological thinning method of [PK99] (92 seconds). Using the potential field method of [CSYB05a] (16 minutes). Our result, computed in 3.1 seconds.*

- approach is very fast and can construct skeleton in few seconds

On-the-fly Curve-Skeleton Computation



www.youtube.com/watch?v=zIPohvB5vHI

MEAN CURVATURE SKELETONS

Mean Curvature Skeletons

Tagliasacchi et.al., SGP 2012

- Iterated mesh contraction via Mean Curvature Flow
- Final conversion to curve skeleton via Edge Collapse

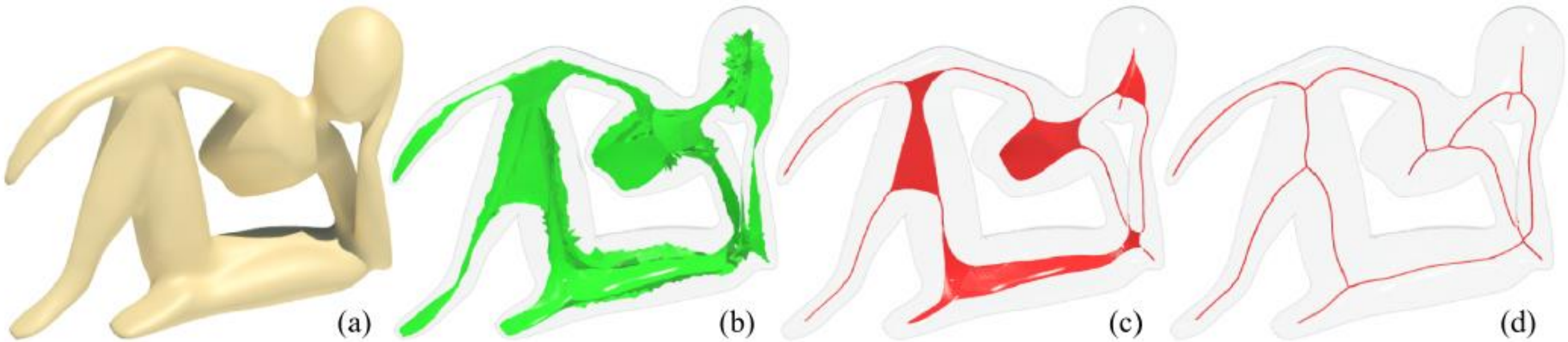


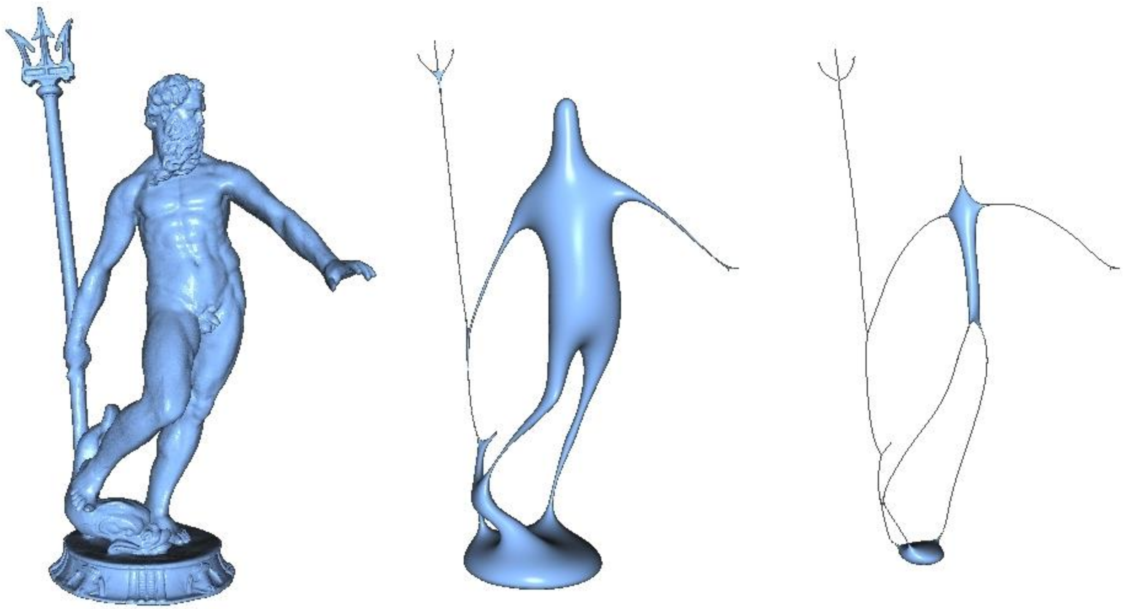
Figure 1: *Given a watertight surface (a), the well-known medial axis transform (b) often produces too complex of a structure to be of practical use. Our skeletonization algorithm can produce intermediate* meso-skeletons *(c), which contain medial sheets where needed and curves where appropriate, while converging to a* medially centered *curve skeleton output (d).*

Mean Curvature Flow

- Describes the motion of a surface that tries to minimize its tension (e.g. oil drop on water)
- Can be expressed with the differential equation

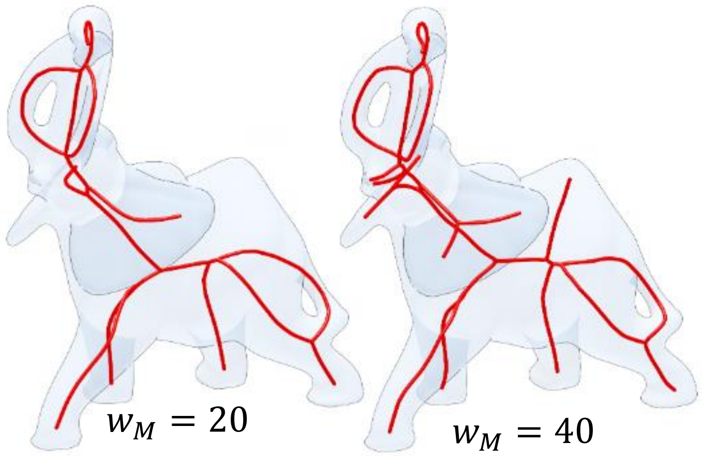
$$\dot{S} = -\frac{\kappa_1 + \kappa_2}{2}\,\boldsymbol{n}$$

for the principal curvatures κ_1, κ_2 and surface normal $\boldsymbol{n}$



Medial Skeletonization Flow

- The vertex locations $V^t = (v_i^t)$ are moved to the curve skeleton by energy minimization in each iteration:
  $V^{t+1} = \arg\min_{X=(x_i)} \left(E_{smooth}(X) + E_{velocity}(X) + E_{medial}(X)\right)$
- $E_{smooth}(X) = w_L \cdot \|LX\|^2$
- $E_{velocity}(X = (x_i)) = w_H \cdot \sum_i \left\|x_i - v_i^t\right\|^2$
- $E_{medial}(X = (x_i)) = w_M \cdot \sum_i \|x_i - \mu(v_i)\|^2$
  - with the weights $w_L = 1$, $w_H = 20$, $w_M = 40$.
  - L is the matrix representation of MCF via Laplacian.
  - $\mu(\cdot)$ maps each vertex to its associated Voronoi pole.
- Remeshing is done after every iteration.
- After convergence, shortest edges are collapsed to produce a 1D curve skeleton.



MCF Skeletons - Results

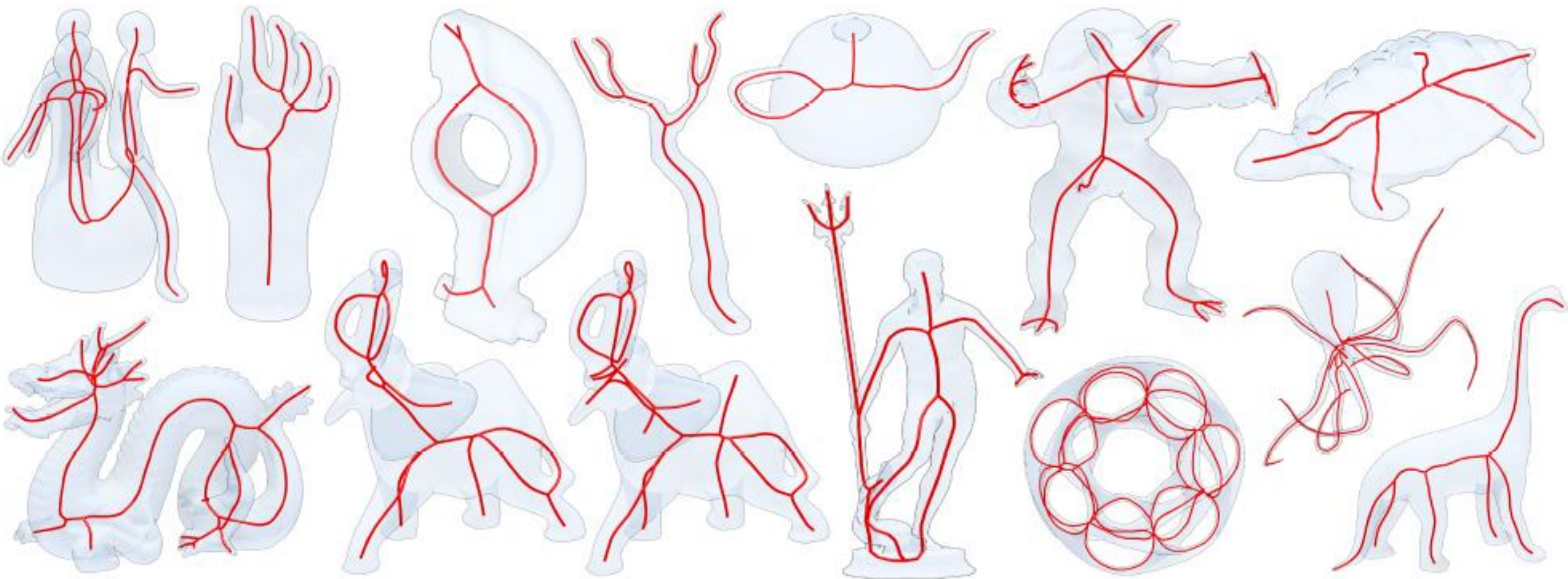


Figure 6: *A gallery of models that have been skeletonized by our algorithm. On each model the algorithm took less than a minute to produce a skeleton, which was produced with the parameters* $\{w_L = 1, w_H = 20, w_M = 40, \varepsilon = .002 * bbox.diag()\}$, *with the exception of the elephant model, where we illustrate how lowering the parameter* $w_M = 20$ *results in a coarser skeleton.*

References

- Tamal K. Dey and Jian Sun. 2006. Defining and computing curve-skeletons with medial geodesic function. In *Proceedings of the fourth Eurographics symposium on Geometry processing* (SGP '06). Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 143-152.
- Nicu D. Cornea, Deborah Silver, and Patrick Min. 2007. Curve-Skeleton Properties, Applications, and Algorithms. *IEEE Transactions on Visualization and Computer Graphics* 13, 3, 530-548
- Andrei Sharf, Thomas Lewiner, Ariel Shamir, Leif Kobbelt: On-the-fly Curve-skeleton Computation for 3D Shapes, Eurographics 2007, pages 323-328
- A. Tagliasacchiy, I. Alhashim, M. Olson, H. Zhang, Mean Curvature Skeletons, SGP 2012, Code: https://code.google.com/p/starlab-mcfskel