3D Model Segmentation

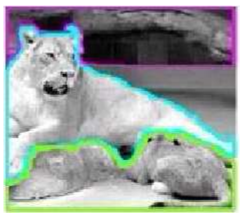
Philip Shilane

- Overview
- Interactive segmentation
  - Scissor tools
  - VRML structure
- Automatic Segmentation
  - Decomposing into convex shapes
    - Polygon Triangulation
    - Space Sweep
  - Surface decomposition
    - Flood and Retract
    - Watershed segmentation
    - K-Means clustering
    - Fuzzy clustering

Decomposition

Applies in different domains:

- Images – "segmentation"



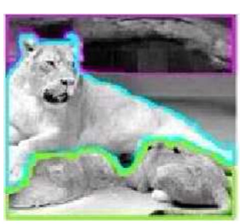
Philip Shilane

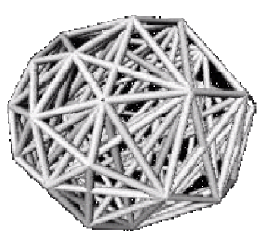
Tal

Decomposition

Applies in different domains:

- Images – "segmentation"
- Polyhedra – "triangulation" or "convex pieces"





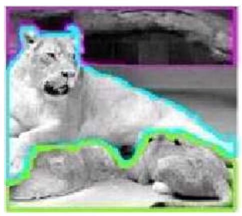
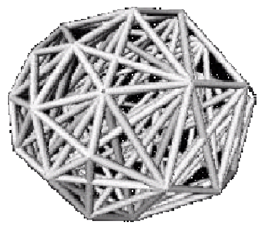
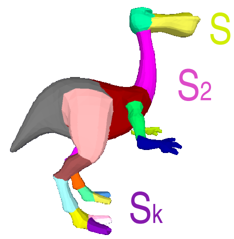
Philip Shilane

Tal

Decomposition

Applies in different domains:

- Images – "segmentation"
- Polyhedra – "triangulation" or "convex pieces"
- Meshes – "decomposition"

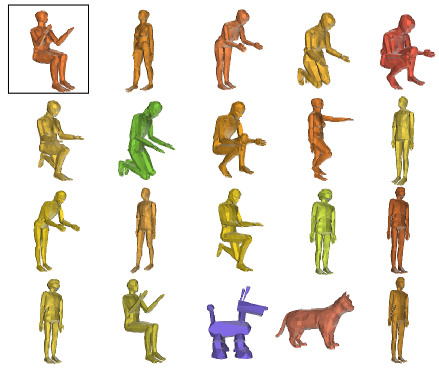




Applications

- Shape-based retrieval
- Metamorphosis
- Simplification
- Compression
- Collision detection
- Control skeleton extraction

Applications

- Shape-based retrieval
  - (Zuckerberger)
- Metamorphosis
- Simplification
- Compression
- Collision detection
- Control skeleton extraction

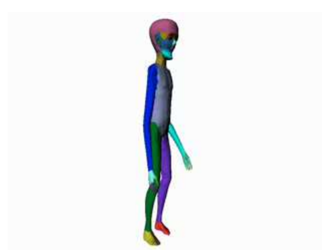


Philip Shilane

Tal

Applications

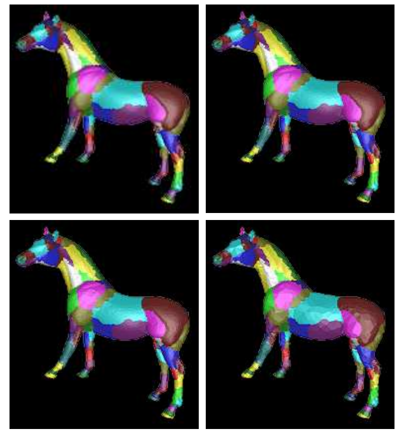
- Shape-based retrieval
- Metamorphosis
  - (Shlafman)
- Simplification
- Compression
- Collision detection
- Control skeleton extraction



Philip Shilane

Tal

Applications



- Shape-based retrieval
- Metamorphosis
- Simplification
  - (Zuckerberger)
- Compression
- Collision detection
- Control skeleton extraction

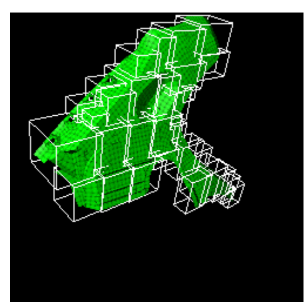
39,697 faces

7,936 faces

Philip Shilane

Tal and Zuckerberger

Applications



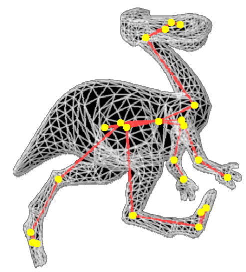
- Shape-based retrieval
- Metamorphosis
- Simplification
- Compression
- Collision detection
- Control skeleton extraction

Philip Shilane

Tal and Frisch

Applications

- Shape-based retrieval
- Metamorphosis
- Simplification
- Compression
- Collision detection
- Control skeleton extraction
  - (Katz)



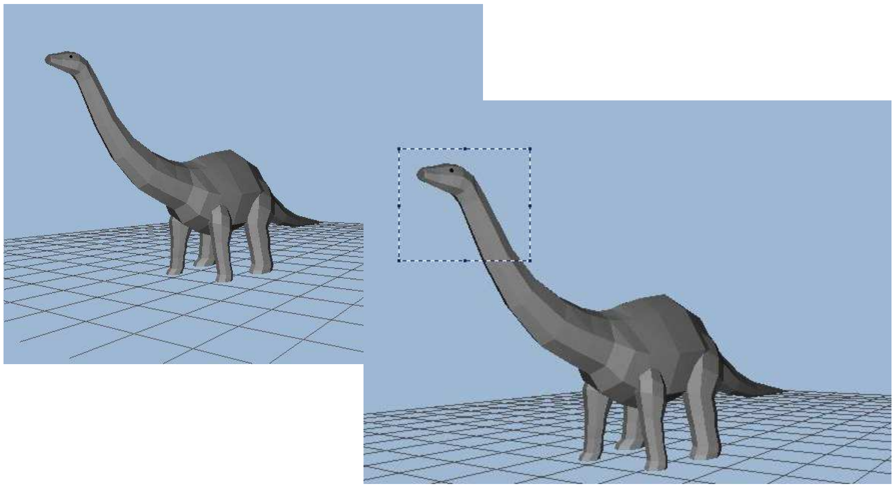
Philip Shilane

Tal

- Overview
- Interactive segmentation
  - Scissor tools
  - VRML structure
- Automatic Segmentation
  - Decomposing into convex shapes
    - Polygon Triangulation
    - Space Sweep
  - Surface decomposition
    - Flood and Retract
    - Watershed segmentation
    - K-Means clustering
    - Fuzzy clustering

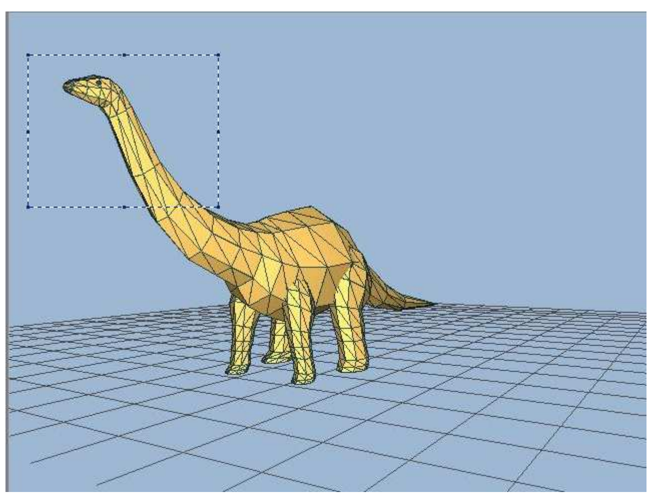
Philip Shilane

Scissor Tools



Philip Shilane

Scissor Tools

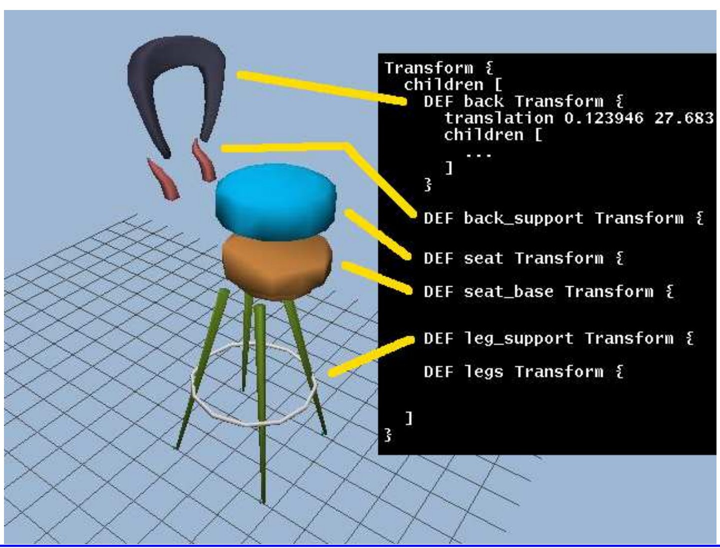


Philip Shilane

- Overview
- Interactive segmentation
  - Scissor tools
  - VRML structure
- Automatic Segmentation
  - Decomposing into convex shapes
    - Polygon Triangulation
    - Space Sweep
  - Surface decomposition
    - Flood and Retract
    - Watershed segmentation
    - K-Means clustering
    - Fuzzy clustering

Philip Shilane

VRML Segmentation

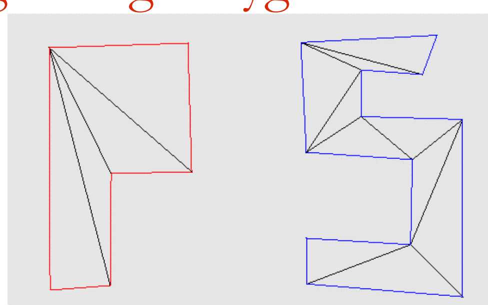


Philip Shilane

Min

- Overview
- Interactive segmentation
  - Scissor tools
  - VRML structure
- Automatic Segmentation
  - Decomposing into convex shapes
    - Polygon Triangulation
    - Space Sweep
  - Surface decomposition
    - Flood and Retract
    - Watershed segmentation
    - K-Means clustering
    - Fuzzy clustering

Triangulating Polygons



Triangulated by ear removal algorithm

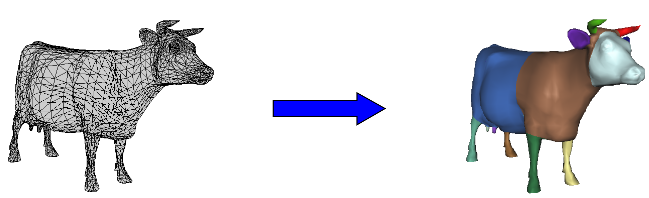
O(n log n) Tarjan 1978 and others, allowing for holes

For polygons without holes:
O(n log log n) Hoffman, Tarjan, etc. 1986
O(n log* n) Clarkson, Tarjan, etc. 1989
O(n) Chazelle 1990

image from Paul Calamia

- Overview
- Interactive segmentation
  - Scissor tools
  - VRML structure
- Automatic Segmentation
  - Decomposing into convex shapes
    - Polygon Triangulation
    - Space Sweep
  - Surface decomposition
    - Flood and Retract
    - Watershed segmentation
    - K-Means clustering
    - Fuzzy clustering

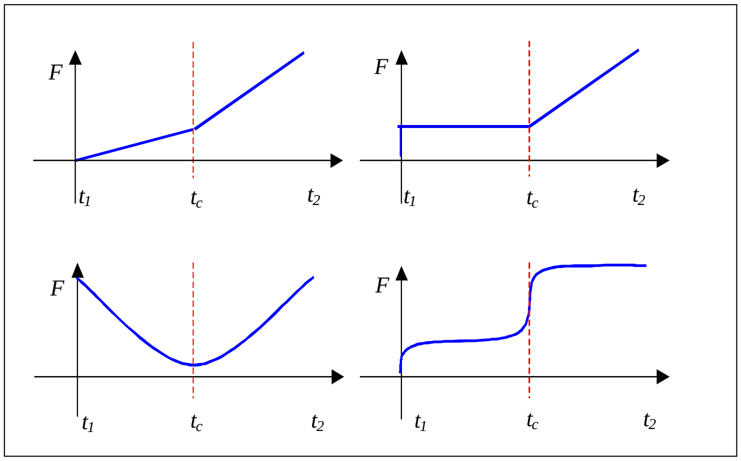
Space Sweep, Xuetao Li, et al



$$O = \{C_i \mid C_i = \bigcup_{t=t_{ia}}^{t_{ib}} G(t), \quad i = 1, \dots, n\}$$

Li

Defining Critical Points



Philip Shilane

Li

Critical Point & Component

Critical Point

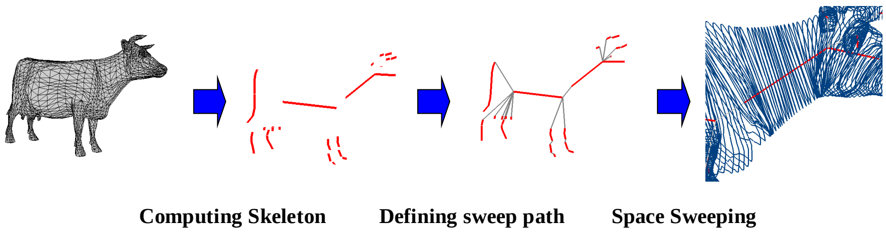
$$\left(F^{(n)}(t) = 0 \text{ and } F^{(n)}(t-\varepsilon)\cdot F^{(n)}(t+\varepsilon) < 0\right) \text{ or } \quad T(t) = 0$$

Component

$$C = \bigcup_{t=t_a}^{t_b} G(t), \text{ such that there is no critical point in } (t_a, t_b)$$

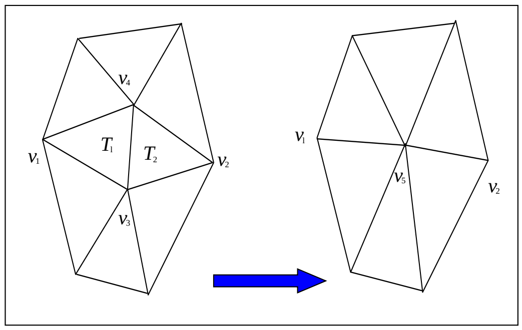
Philip Shilane

Sweeping up skeletons



Simplification

Associated Triangle List (*ATL*)

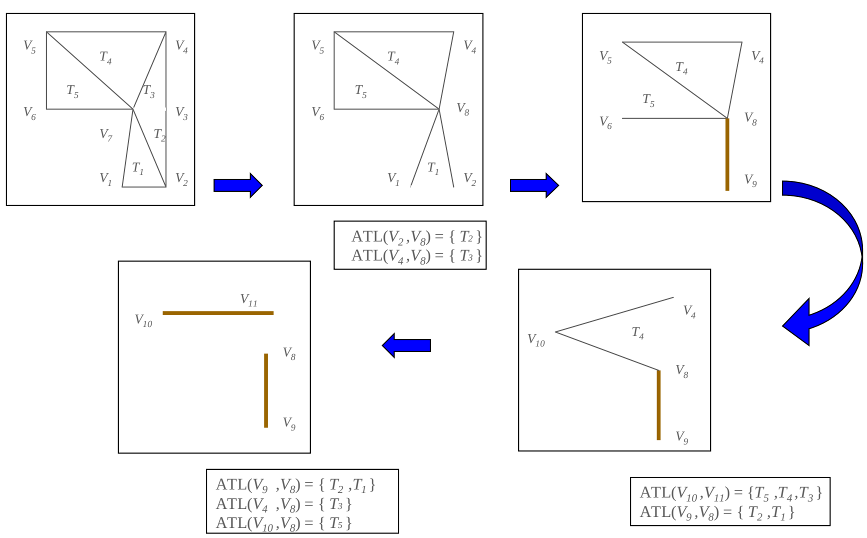


$ATL(v_1, v_5) = \{T_1\}$

$ATL(v_2, v_5) = \{T_2\}$

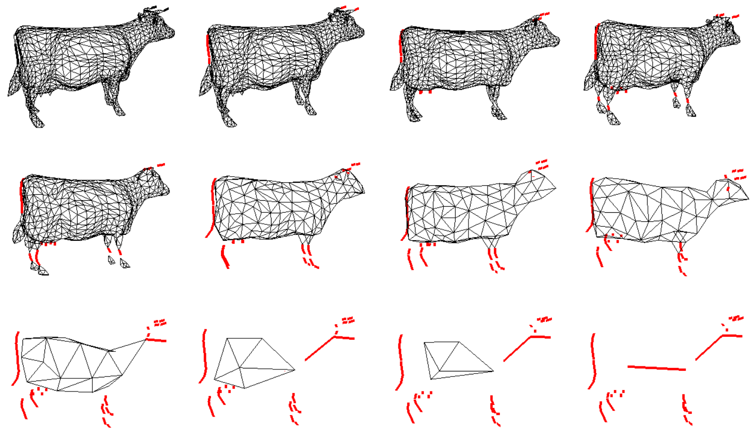
Edge collapse

2D Example



3D Example

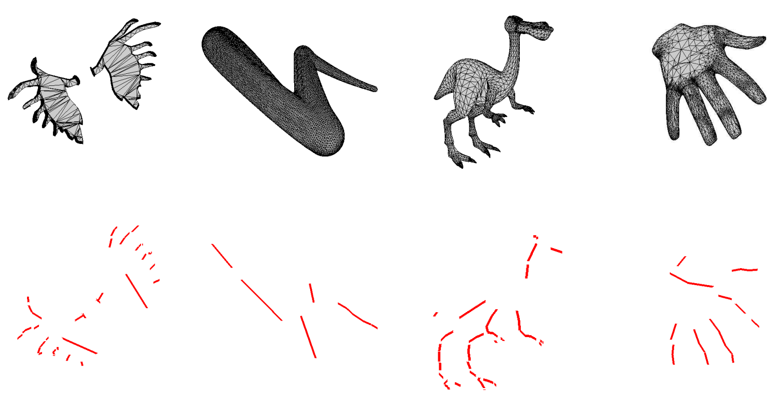
O(n log n) where the n vertices have a bounded number of edges



Philip Shilane

Li

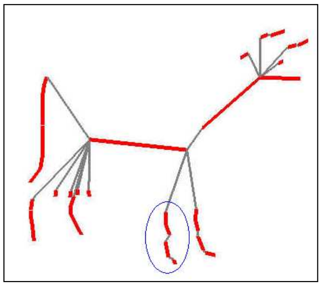
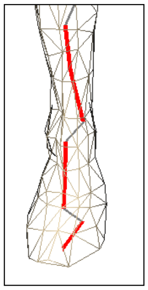
Skeleton of Polygon Mesh



Philip Shilane

Li

Sweeping the skeleton

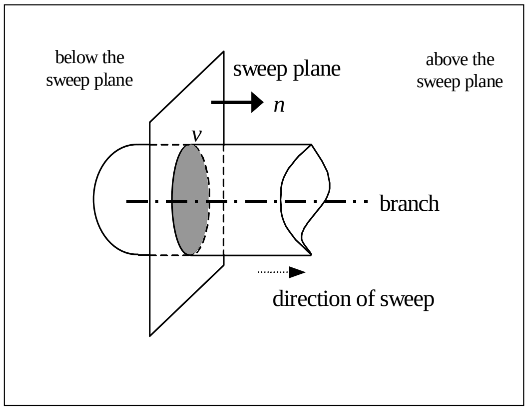
Branches swept in order of
increasing surface area of ATL

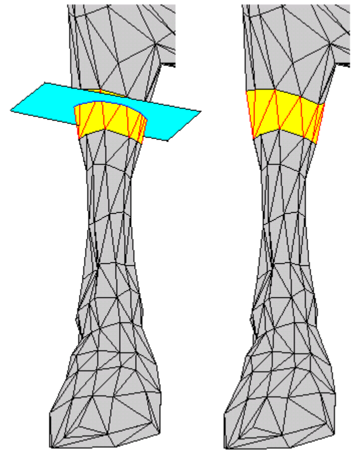
$$A(P) = (\text{length of branch}) * \frac{Perimeter(u) + Perimeter(v)}{2}$$

Sweep Operation

- Compute the cross sections
- Check if critical point is reached
    - If reached, extract component.
    - If not, move the sweep plane to the next position

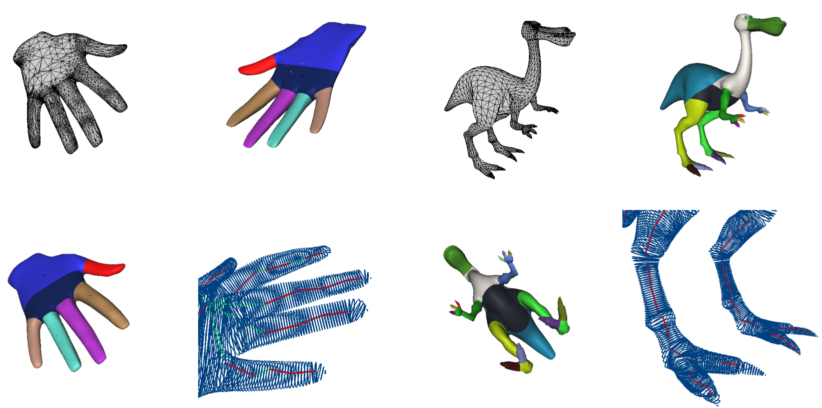
Sweep Plane





Philip Shilane   Li

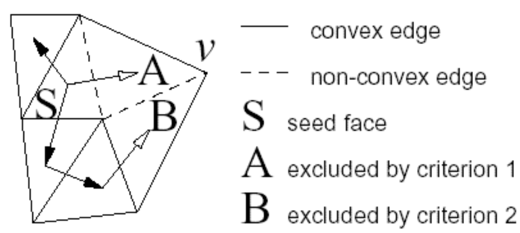
Space Sweep Results



Philip Shilane   Li

- Overview
- Interactive segmentation
  - Scissor tools
  - VRML structure
- Automatic Segmentation
  - Decomposing into convex shapes
    - Polygon Triangulation
    - Space Sweep
  - Surface decomposition
    - Flood and Retract
    - Watershed segmentation
    - K-Means clustering
    - Fuzzy clustering

Flooding



1. Create the dual graph, where nodes represent faces and edges join nodes with adjacent faces.
2. Start at a seed node.
3. Collect allowable nodes.

Stopping Criteria
1 Local failure: edge exhibits non-convexity
2 Global failure: locally convex, but not on the convex hull

image from Luv Kohli

Flood and Retract, Chazelle et al

Global failures occur with twisted shapes.

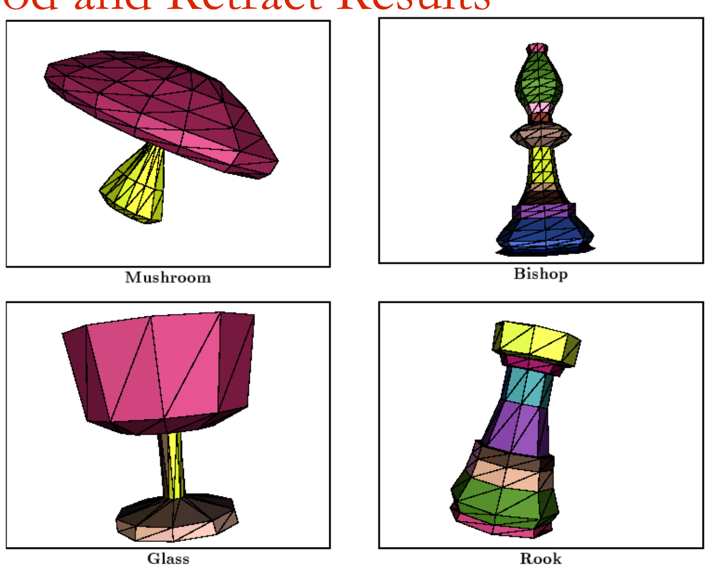
Otherwise greedy flooding would be optimal.

1. Flood the surface by covers, allowing overlap of faces.
2. Retract each cover that lies within the boundary of another patch as much as possible, so that the patch remains connected.

Philip Shilane

image from www.treehopper.com/ index.php?IID=4

Flood and Retract Results



Philip Shilane

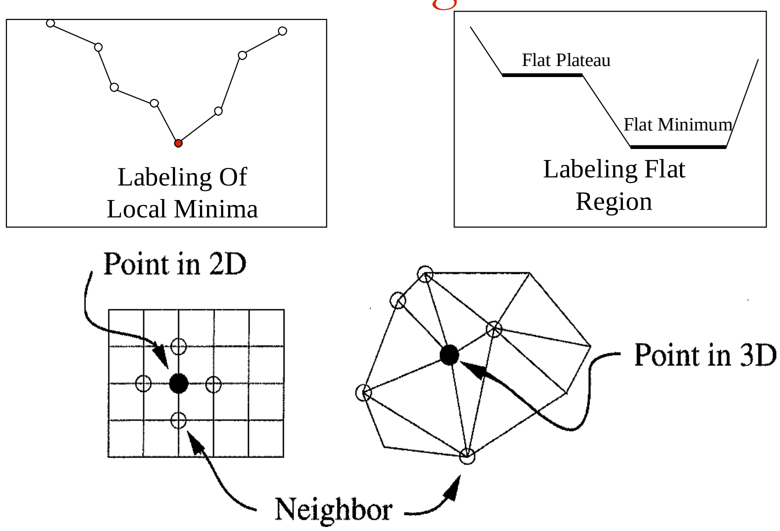
Chazelle

- Overview
- Interactive segmentation
  - Scissor tools
  - VRML structure
- Automatic Segmentation
  - Decomposing into convex shapes
    - Polygon Triangulation
    - Space Sweep
  - Surface decomposition
    - Flood and Retract
    - Watershed segmentation
    - K-Means clustering
    - Fuzzy clustering

Watershed Segmentation, Mangan et al.

1. Compute the curvature (or some other height function) at each vertex.
2. Find the local minima and assign each a unique label.
3. Find each flat area and classify it as a minimum or plateau.
4. Loop through the plateaus and allow each one to descend until a labeled region is encountered.
5. Allow all remaining unlabeled vertices to descend and join a labeled region.
6. Merge regions whose watershed depth is below a threshold.

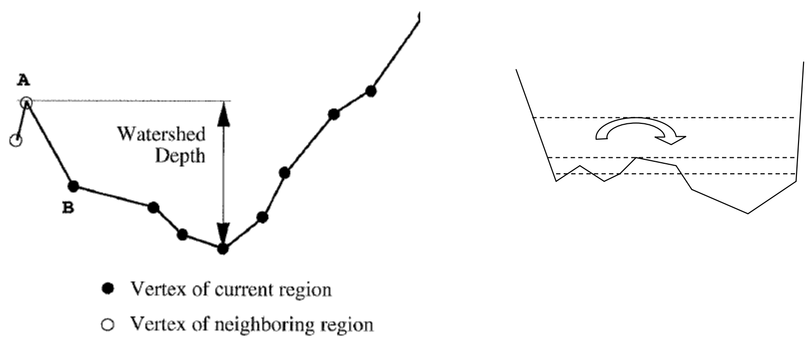
Watershed Flooding



Philip Shilane

Mohandas and Mangan

Merging regions



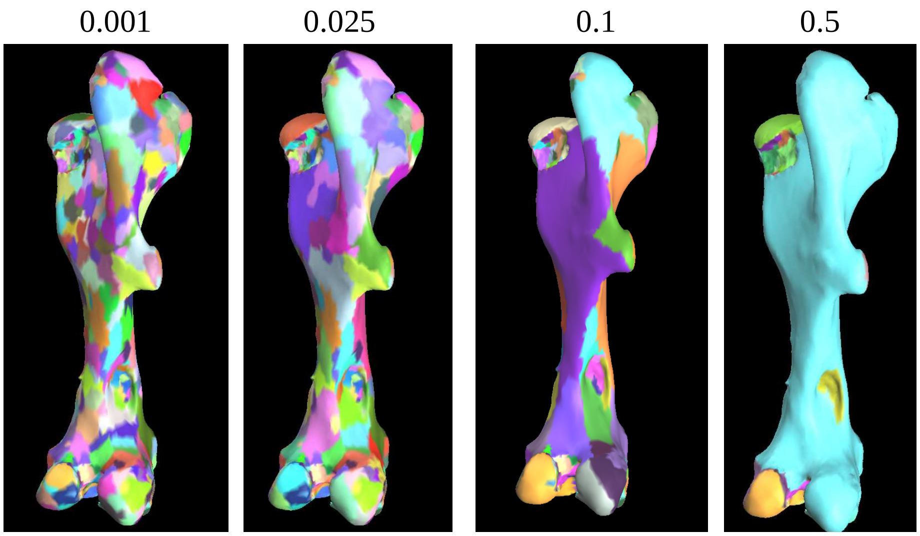
Define the depth of a region by its lowest vertex and lowest boundary vertex.

Merge adjacent regions with shallow depths.

Philip Shilane

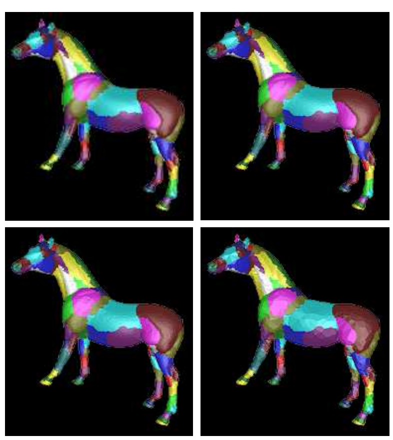
Mohandas and Mangan

Varying Depth Threshold



Mohandas

Simplifying Segments



Simplified from 39,697 to 7,936 faces while maintaining segments.

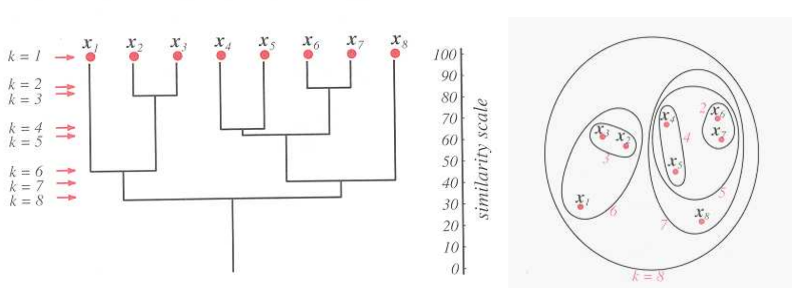
Zuckerberger

- Overview
- Interactive segmentation
  - Scissor tools
  - VRML structure
- Automatic Segmentation
  - Decomposing into convex shapes
    - Polygon Triangulation
    - Space Sweep
  - Surface decomposition
    - Flood and Retract
    - Watershed segmentation
    - K-Means clustering
    - Fuzzy clustering

Hierarchical Clustering

Agglomerative clustering
joining clusters, bottom up

Divisive clustering
splitting clusters, top down



Duda

K-Means Clustering

- Initialize k reference vectors (or means) μ_k to k of the data points.
- E – step: Each data point x_i is assigned to one of the k regions by

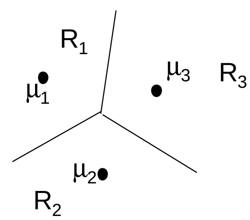
$$k(x_i) = \arg\min_k \|x_i - \mu_k\|^2$$

defines regions

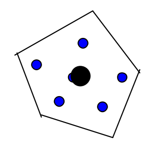
$$R_k = \{\ x \mid \ \|x - \mu_k\| < \|x - \mu_j\|\ \}$$

- M- step: Move μ_k to centroid of Rk

$$\mu_k = \frac{1}{N_k} \sum_{x_i \varepsilon R_k} x_i, \qquad N_k = \sum_{x_i \varepsilon R_k} 1$$



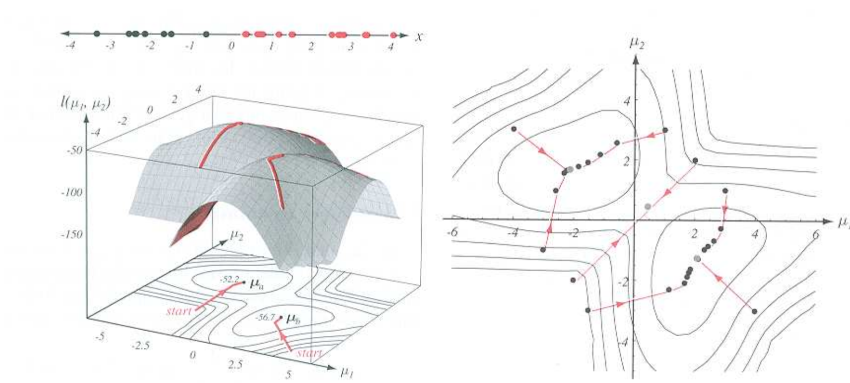
Voronoi tesselation



Philip Shilane

Leen

K-Means Example



Philip Shilane

Duda

Distance Metrics

Minkowski metric

$$dist(a,b)=\left(\sum_{i=1}^{d}|a_i-b_i|^q\right)^{1/q}$$

cosine distance

$$dist(a,b)=\frac{a^t b}{\|a\|\|b\|}$$

shared binary features

$$dist(a,b)=\frac{a^t b}{d}$$

Mahalanobis distance

$$dist(a,\mu)=(a-\mu)^t\Sigma^{-1}(a-\mu)$$

Error Criteria

Sum of Squared Error

$$Error=\sum_{i=1}^{numCat}\sum_{x\in Cat_i}dist(x,\mu_i)$$

Variance

$$Error=\frac{1}{2}\sum_{i=1}^{numCat}\frac{1}{n_i}\sum_{a\in Cat_i}\sum_{b\in Cat_i}dist(a,b)$$

K-Means Surface Decomposition, Shlafman et al

- Initialize cluster centers with faces as far apart as possible
- Distance Metric:
  - F_1, F_2 adjacent:

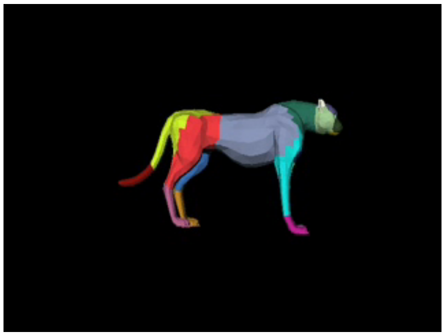
    $$Dist(F_1,F_2)=(1-\delta)(1-\cos^2(\alpha))+\delta Phys\_Dist(F_1,F_2)$$

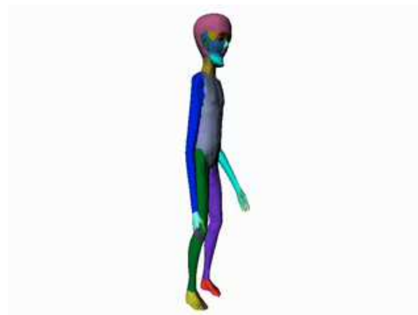
  - F_1, F_2 non-adjacent:

    $$Dist(F_1,F_2)=\min_{F_3\neq F_1,F_2}(Dist(F_1,F_3)+Dist(F_3,F_2))$$

- Error Metric: $\sum_{p\in patches}\sum_{f\in p}Dist(f,\mathrm{Rep}(p))$

K-Means Segmentation Results





Philip Shilane

Shlafman and Tal

- Overview
- Interactive segmentation
  - Scissor tools
  - VRML structure
- Automatic Segmentation
  - Decomposing into convex shapes
    - Polygon Triangulation
    - Space Sweep
  - Surface decomposition
    - Flood and Retract
    - Watershed segmentation
    - K-Means clustering
    - Fuzzy clustering

Philip Shilane

Hierarchical Fuzzy Clustering, Katz et al

- Hierarchical
- Optimizes both boundaries and components
- Avoids over-segmentation

Philip Shilane

Tal

Benefits

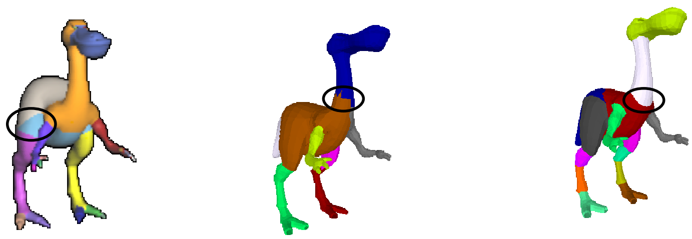
- Hierarchical
- Optimizes both boundaries and components
- Avoids over-segmentation

Philip Shilane

Tal

Benefits

- Hierarchical
- Optimizes both boundaries and components
- Avoids over-segmentation

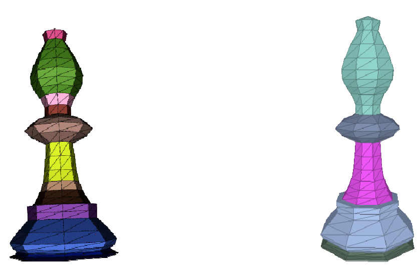


Li et al — Shlafman et al — Katz et al

Philip Shilane — Tal

Benefits

- Hierarchical
- Optimizes both boundaries and components
- Avoids over-segmentation

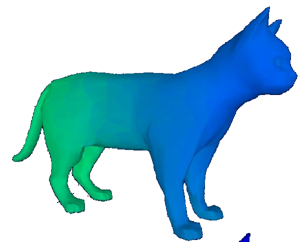


Chazelle et al — Katz et al

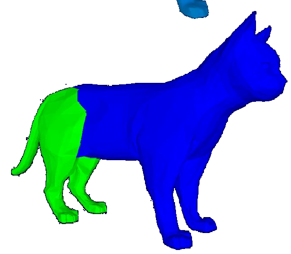
Philip Shilane — Tal

Key idea: two-phase algorithm

1. Find major components with fuzzy boundaries



2. Find exact boundaries



Algorithm outline

1. Construct fuzzy decomposition
   a. Assign distances to pairs of faces
   b. Assign probabilities of belonging to patches
   c. Compute a fuzzy decomposition
2. Construct exact boundaries

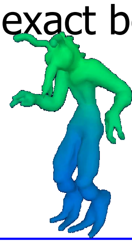
Algorithm outline

1. Construct fuzzy decomposition
   a. Assign distances to pairs of faces
   b. Assign probabilities of belonging to patches
   c. Compute a fuzzy decomposition
2. Construct exact boundaries

Philip Shilane

Algorithm outline

1. Construct fuzzy decomposition
   a. Assign distances to pairs of faces
   b. Assign probabilities of belonging to patches
   c. Compute a fuzzy decomposition
2. Construct exact boundaries

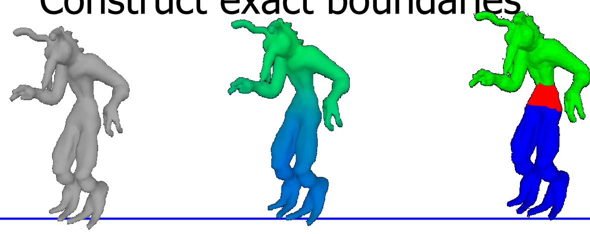


Philip Shilane

Tal

Algorithm outline

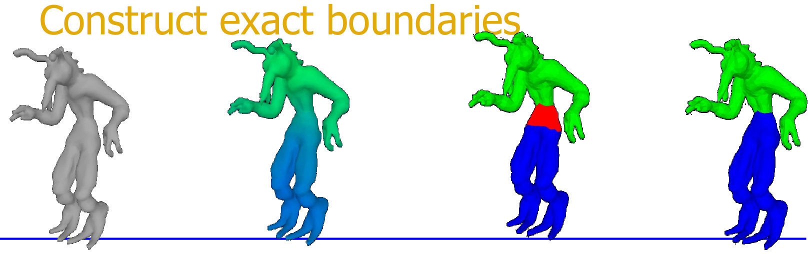
1. Construct fuzzy decomposition
   a. Assign distances to pairs of faces
   b. Assign probabilities of belonging to patches
   c. Compute a fuzzy decomposition
2. Construct exact boundaries



Philip Shilane

Algorithm outline

1. Construct fuzzy decomposition
   a. Assign distances to pairs of faces
   b. Assign probabilities of belonging to patches
   c. Compute a fuzzy decomposition
2. Construct exact boundaries



Philip Shilane

Tal

Distance function

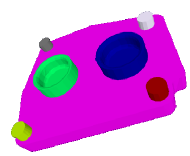
Distant faces less likely to belong to same patch

- Initially, adjacent faces:

$$Dist(f_i, f_j) = (1-\delta)\cdot\frac{AngDist(\alpha_{ij})}{avg(AngDist)} + \delta\cdot\frac{GeodDist(f_i, f_j)}{avg(GeodDist)}$$

$$AngDist(\alpha_{ij}) = \eta(1-\cos\alpha_{ij})$$

- Final distances - shortest paths



Probabilities

$$P_B(f_i) = \frac{Dist(f_i, REP_A)}{Dist(f_i, REP_A) + Dist(f_i, REP_B)} = \frac{a_{f_i}}{a_{f_i} + b_{f_i}}$$

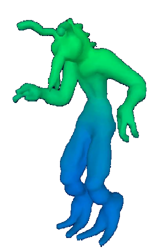
Properties

I. $\forall a_{f_i} < b_{f_i},\ P_B(f_i) < 0.5$

II. $\forall a_{f_i} > b_{f_i},\ P_B(f_i) > 0.5$

III. $\forall a_{f_i} = b_{f_i},\ P_B(f_i) = 0.5$

IV. $P_B(f_i) = 1 - P_A(f_i)$



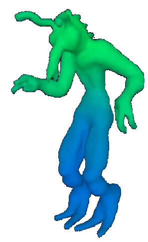
Fuzzy K-means

Goal: optimize $F = \sum_p \sum_f \Pr(f \in patch(p)) \cdot Dist(f, p))$

1. Initialization - select set of representatives Vk
2. Compute probabilities
3. Re-compute the set of representatives $V_{k'}$

$$REP_B = \min_f \sum_{fi} P_B(f_i) \cdot Dist(f, f_i)$$

4. If V_k is sufficiently different from $V_{k'}$, set $V_k \longleftarrow V_{k'}$ and go back to 2



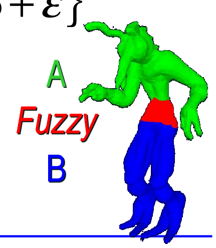
Philip Shilane

Tal

Fuzzy decomposition

The surface is decomposed into *A, B, Fuzzy*

$$A = \{f_i \mid P_B(f_i) < 0.5 - \varepsilon\}$$
$$B = \{f_i \mid P_B(f_i) > 0.5 + \varepsilon\}$$
$$Fuzzy = \{f_i \mid 0.5 - \varepsilon \le P_B(f_i) \le 0.5 + \varepsilon\}$$



Philip Shilane

Tal

Problem: finding boundaries

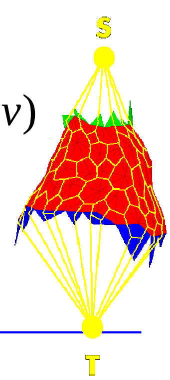
- Given:
  - G=(V,E) the dual graph of the mesh
  - A,B,Fuzzy
- Partition V into $V_{A'}$ and $V_{B'}$ s.t.

I. $V = V_{A'} \cup V_{B'}$
II. $V_{A'} \cap V_{B'} = \phi$
III. $V_A \subseteq V_{A'}, V_B \subseteq V_{B'}$
IV. Good cut!

Algorithm for finding boundaries

- Assign capacities
- Construct a flow network on *Fuzzy*
- Find the minimum cut

$$weight(Cut(V_{A'}, V_{B'})) = \sum_{u \in V_{A'}, v \in V_{B'}} w(u,v)$$



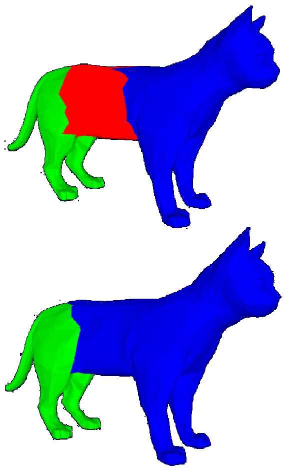
Assigning capacities

Cuts should pass at regions of deep concavities (*Biederman*)

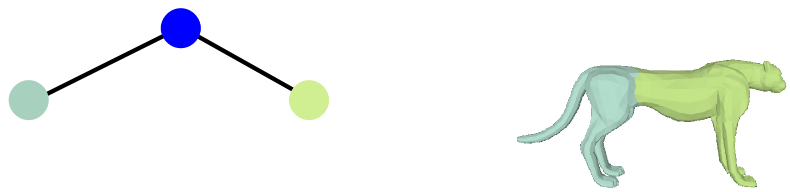
$$Cap(i,j) = \frac{1}{1 + \frac{AngDist(\alpha_{ij})}{avg(AngDist)}}$$

Key idea: recap

1. Find major components using fuzzy clustering
2. Find exact boundaries using minimum cuts



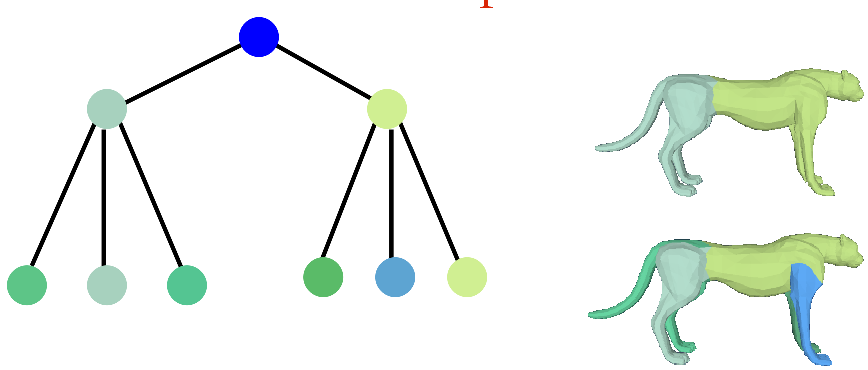
Hierarchical Decomposition



Philip Shilane

Tal

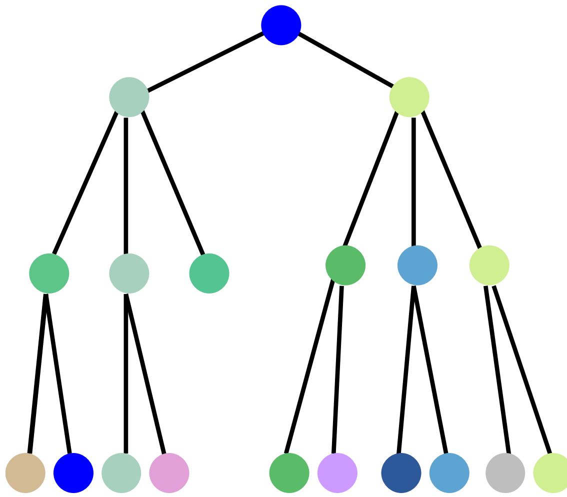
Hierarchical Decomposition

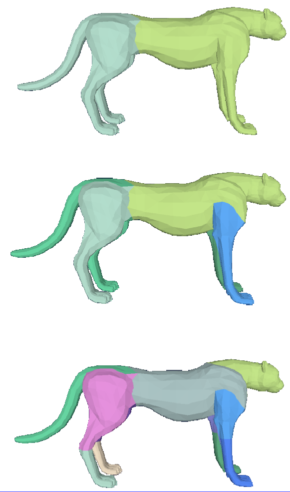


Philip Shilane

Tal

Hierarchical Decomposition





Philip Shilane

Tal

K-way decomposition

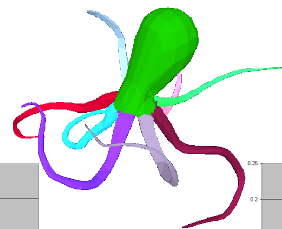
- Determining the number of patches
- Selecting initial representatives
- Assigning probability
- ü Extracting fuzzy areas

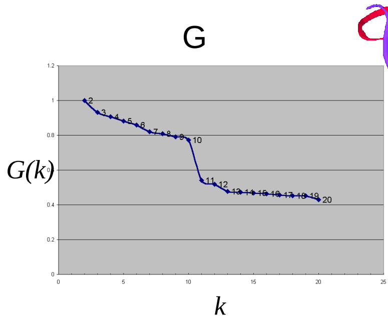
Philip Shilane

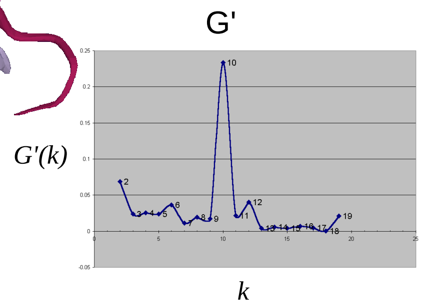
Tal

Determining #patches

$$G(k) = \min_{i<k}(Dist(REP_k, REP_i))$$







K-way decomposition

- Determining the number of patches
- Selecting initial representatives

  The first representative minimizes the distance to all other faces, representing the main body. Other representatives added to be as far apart as possible.
- Assigning probability

$$P_j(f_i) = \frac{\dfrac{1}{Dist(f_i, \mathrm{Rep(j)})}}{\sum_m \dfrac{1}{Dist(f_i, \mathrm{Rep(m)})}}$$

- Extracting fuzzy areas

Hierarchical k-way decomposition

Philip Shilane

Tal

Citations

- Chazelle, B. and Palios, L. 1994. Decomposition algorithms in geometry. In Algebraic Geometry and its Applications, Spring-Verlag, C. C. Bajaj, Ed., 419-447.
- Chazelle, B., Dobkin, D., Shourhura, N., and Tal, A. 1997. Strategies for polyhedral surface decomposition: An experimental study. Computational Geometry: Theory and Applications 7, 4-5, 327-342.
- Li, X., Toon, T., Tan, T., and Huang, Z. 2001. Decomposing polygon meshes for interactive applications. In Proceedings of 2001 symposium on Interactive 3D graphics, 35-42.
- Mangan, A., and Whitaker, R. 1999. Partitioning 3D surface meshes using watershed segmentation. IEEE Transactions on Visualization and Computer Graphics 5, 4, 308-321.
- Shlafman, S., Tal, A., and Katz, S. 2002. Metamorphosis of polyhedral surfaces using decomposition. In Eurographics 2002, 219-228.
- Zuckerberger, E., Tal, A., and Shlafman, S. 2002. Polyhedral surface decomposition with applications. Computers & Graphics 26, 5, 733-743.
- Katz, S., and Tal, A. 2003 Hierarchical Mesh Decomposition using Fuzzy Clustering and Cuts. In SIGGRAPH 2003, ACM Transactions on Graphics, Volume 22 , Issue 3, 954-961.
- Duda, R., Hart, P., and Stork, D. 2000. *Pattern Recognition*. John Wiley & Sons.

Philip Shilane