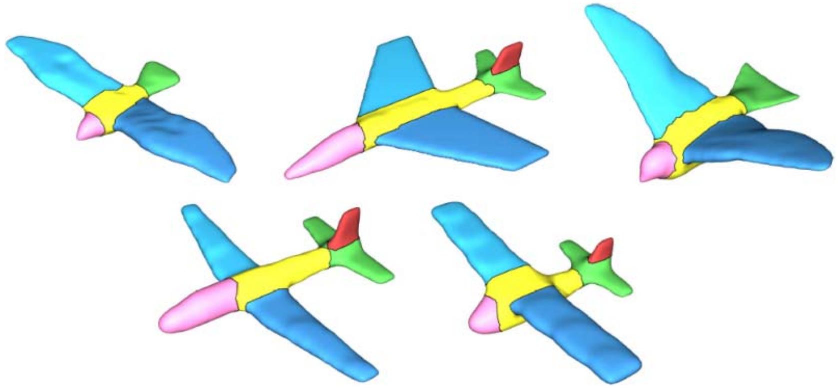
Shape Segmentation



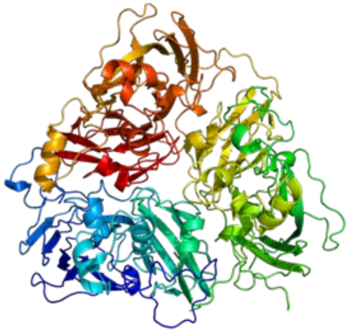
Qixing Huang

3D Shapes

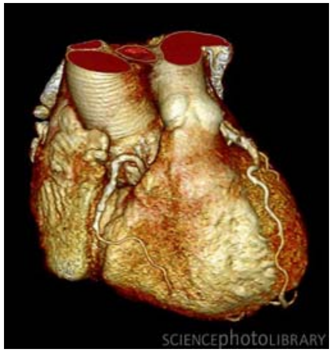
Large repositories of 3D data are becoming available

Shape Modeling

Mechanical CAD



Molecular Biology

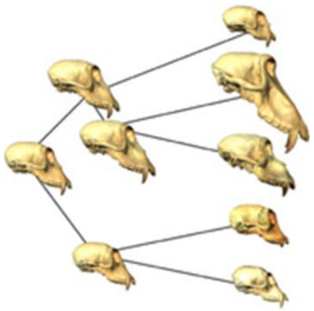


Medicine

Cultural Heritage

Buildings

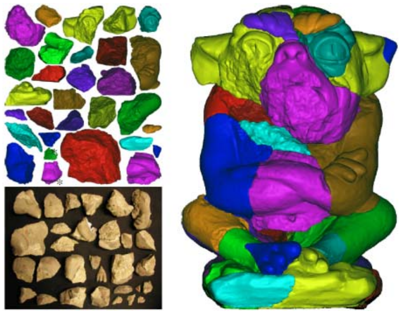
Applications



[*Wiley* et al.05]

Click to learn more about the science behind foldit.

[Cooper et al.10]



[Huang et al.06]

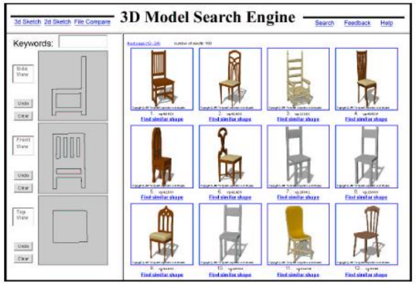
Paleontology

Protein folding

Solving puzzles

[Funkhouser et al.04]

[Gal et al.09]



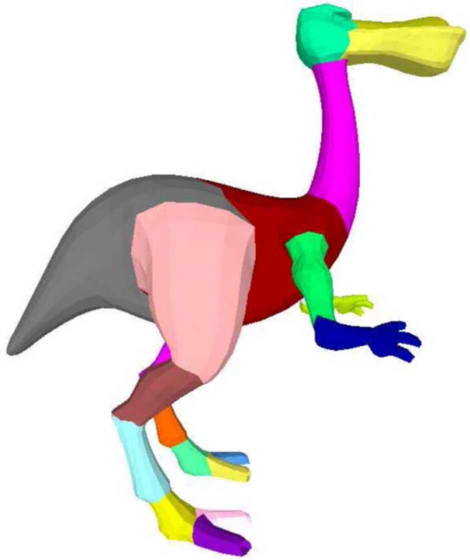
[Funkhouser et al.05]

Modeling & Editing

Product search

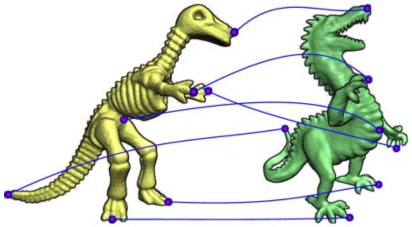
Shape Analysis Tasks

- Design algorithms to extract semantic information from one or a collection of shapes



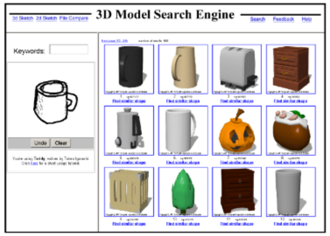
[Karz and Tal 03]

Segmentation



[van Kaick et al. 11]

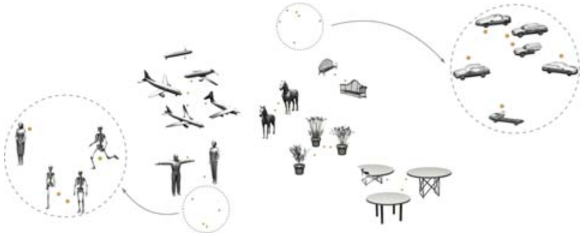
Matching



You're using Teddy, written by Takeo Igarashi. Click here for a short usage tutorial.

[Funkhouser et al. 05]

Retrieval

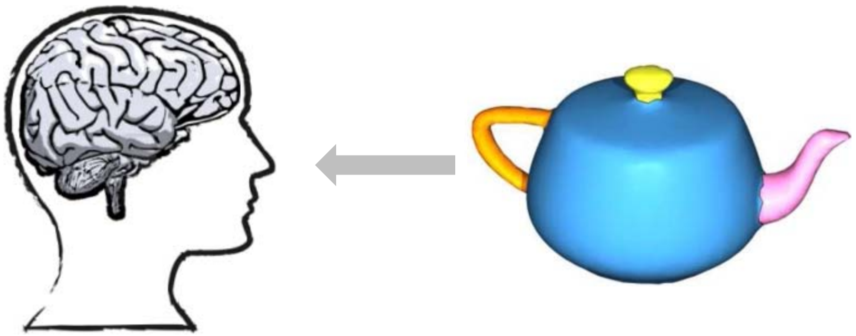


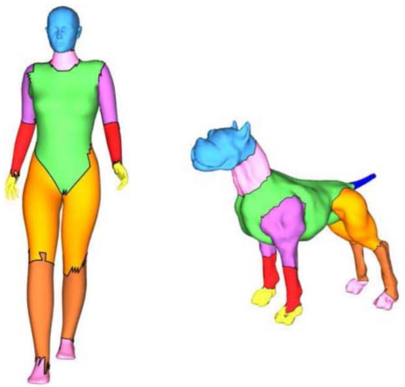
[Mitra et al. 06]

Classification & Clustering

Importance of Shape Segmentation

"How can we decompose a 3D model into parts?"





Psychological research indicates that recognition and shape understanding are based on structural decomposition of the shape into smaller parts [Hoffmann et al. 84,97]

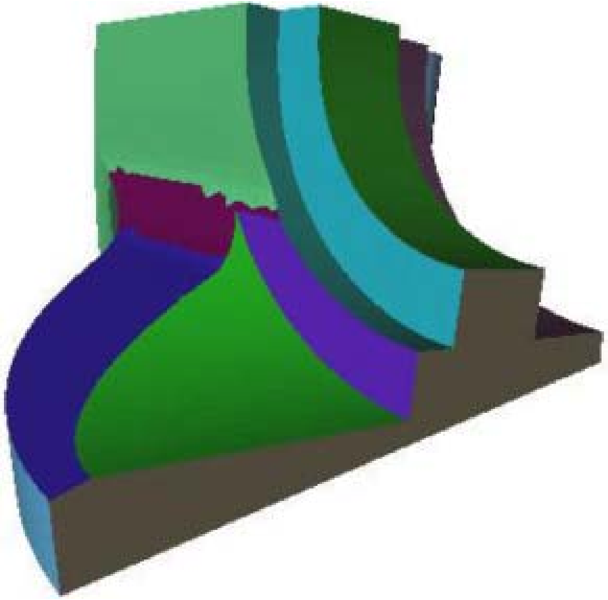
Applications in other shape analysis tasks such as shape matching and shape recognition

Outline

Outline

- Single-shape segmentations
  - Primitive fitting
  - Hierarchical mesh decomposition
  - Princeton segmentation benchmark
- Data-driven shape segmentations
  - Supervised segmentation
  - Joint-shape segmentation
- Conclusion and future directions

Primitive Fitting



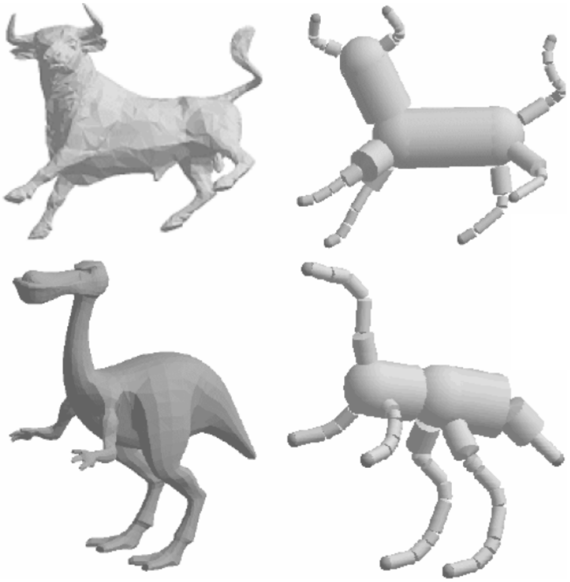
Problem Statement

- Given a mesh M = {V,E,F}, find a disjoint partitioning of M into $M_1,\ldots,M_k$ and a set of (K?) *primitives* $P_1,\ldots,P_k$ such that a *distance* between each primitive P_i to M_i be minimized.

Primitives

- Planes or Cylinders

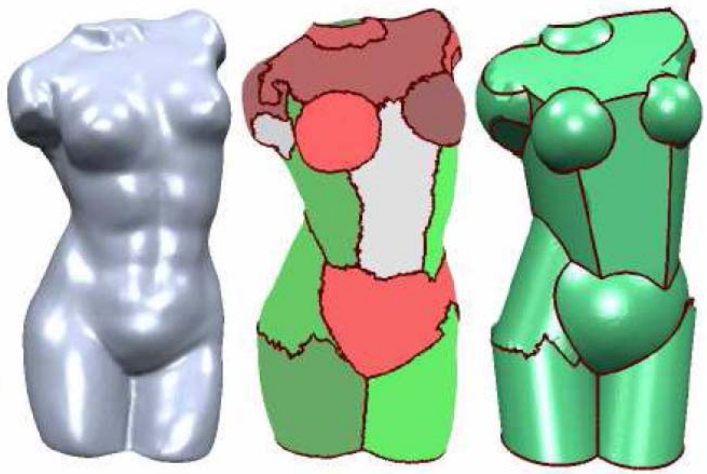
[Cohen-Steiner et al. 04]



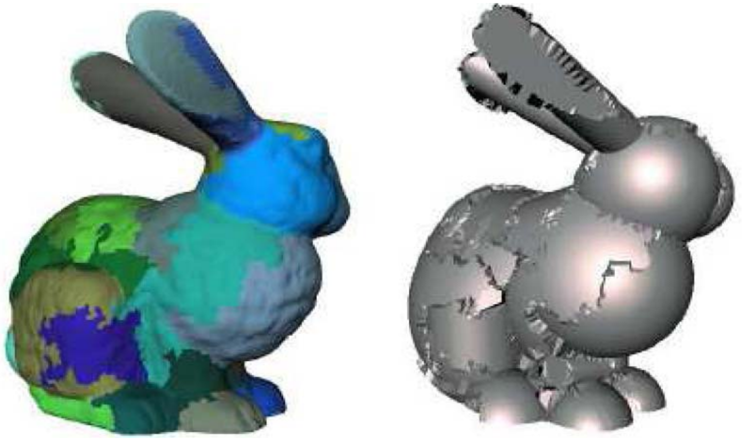
[Raab et al. 04]

Primitives

- Spheres, Hybrid,…



[Wu et al. 05]

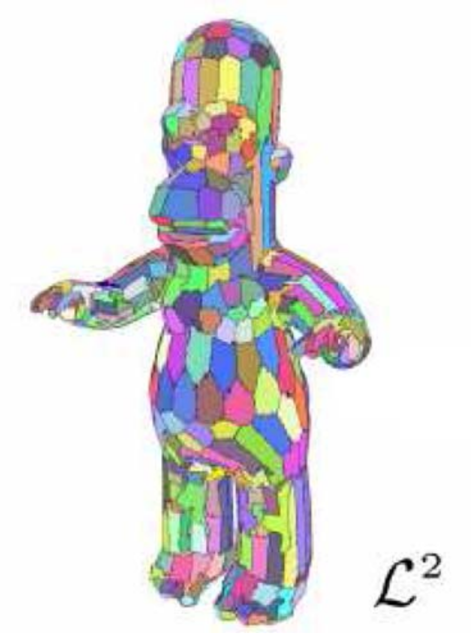


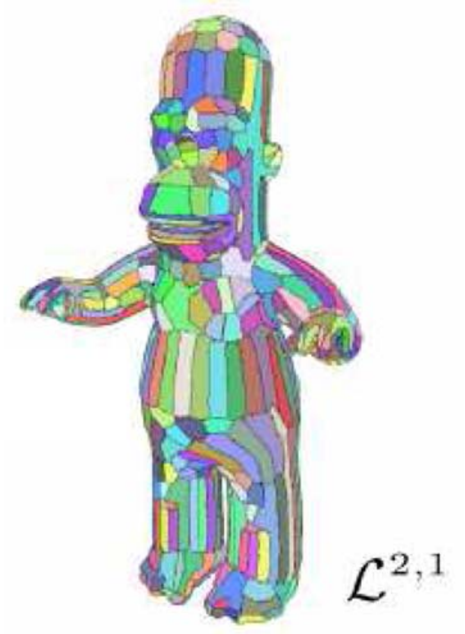
[Attene et al. 06]

Effects of Different Metrics

$$\mathcal{L}^2(\mathcal{R}_i, P_i) = \iint_{x\in\mathcal{R}_i} \|x - \Pi_i(x)\|^2 dx.$$

$$\mathcal{L}^{2,1}(\mathcal{R}_i, P_i) = \iint_{x\in\mathcal{R}_i} \|\mathbf{n}(x) - \mathbf{n}_i\|^2 dx.$$

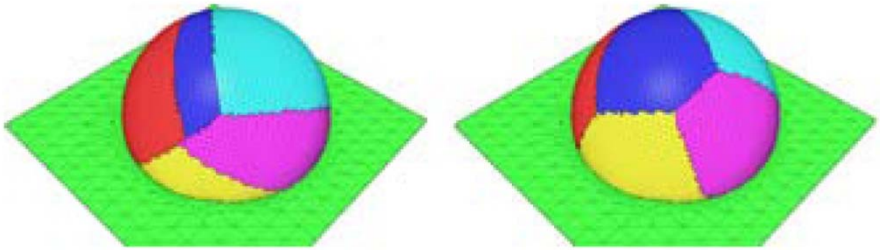


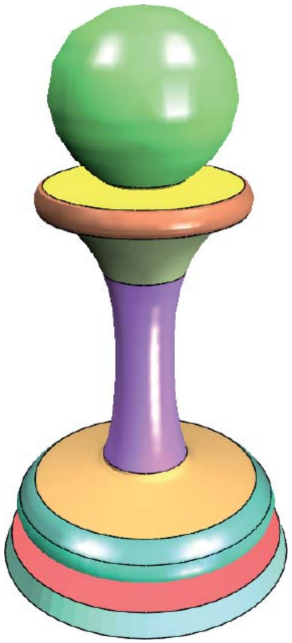


Cohen-Steiner D., Alliez P., Desbrun M.: Variational shape approxiimattiion. ACM Trans.. Graph.. 23,, 3 (2004),, 905–914..

Iterative Lloyd

- RANSAC based initialization
- Alternate between
  - Fitting parameters of each primitive
  - Assigning points to closest patches
- Insert patches

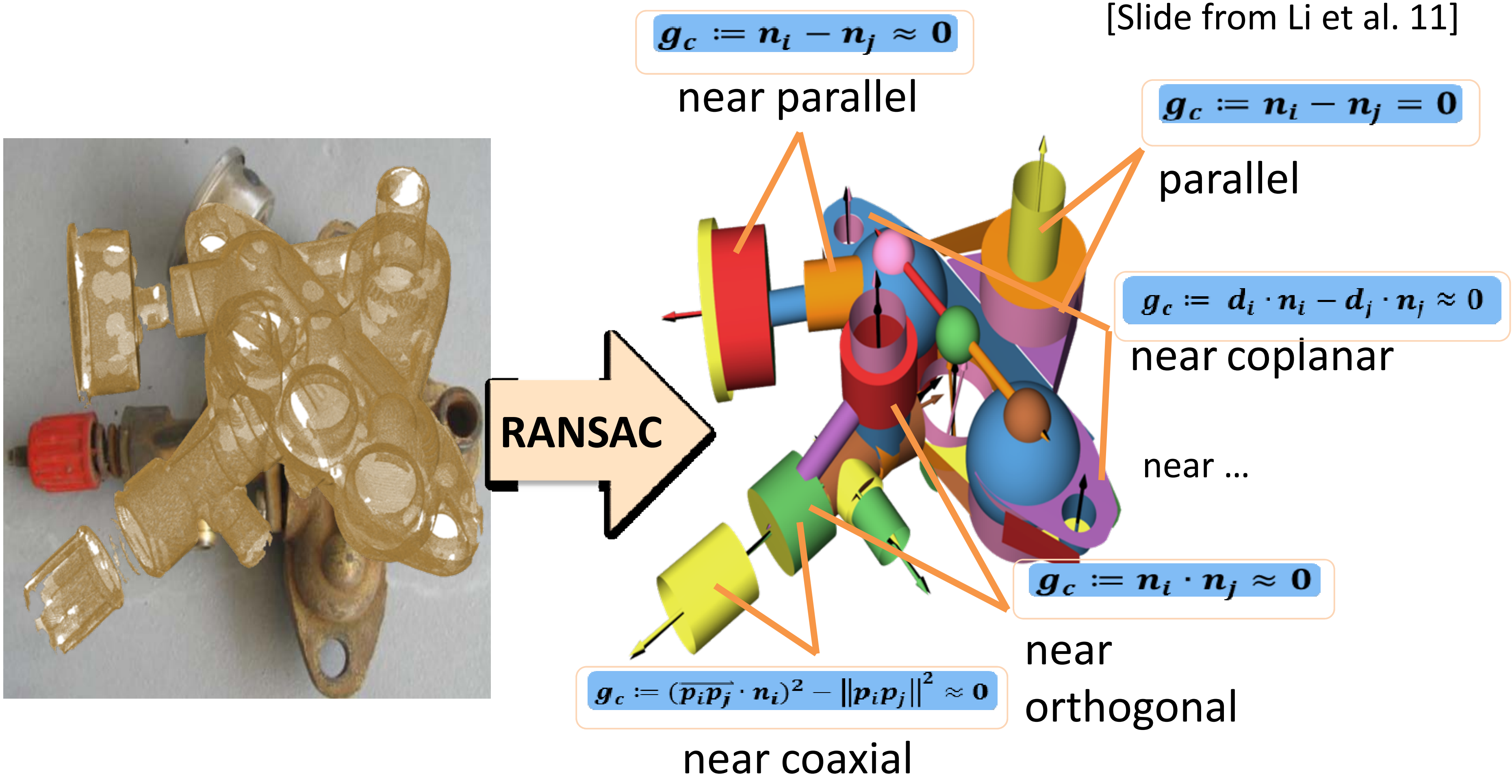




[Yan et al. 06]

Primitive Fitting + Global Relations

[Slide from Li et al. 11]



RANSAC

$g_c := \boldsymbol{n_i} - \boldsymbol{n_j} \approx \boldsymbol{0}$

near parallel

$g_c := \boldsymbol{n_i} - \boldsymbol{n_j} = \boldsymbol{0}$

parallel

$g_c := d_i \cdot n_i - d_j \cdot n_j \approx 0$

near coplanar

near …

$g_c := \boldsymbol{n_i} \cdot \boldsymbol{n_j} \approx \boldsymbol{0}$

near orthogonal

$g_c := (\overrightarrow{p_i p_j} \cdot n_i)^2 - \|p_i p_j\|^2 \approx 0$

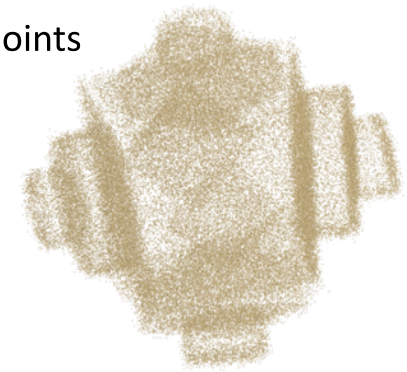
near coaxial

Point Cloud **Initial Primitives**

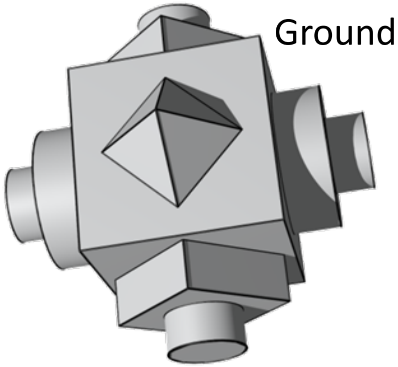
GlobFit: Consistently Fitting Primitives by Discovering Global Relations. Yangyan Li, X. Wu, Y. Chrysanthou, A. Sharf, D. Cohen-Or and Niloy Mitra. Siggraph 2011.

Comparison

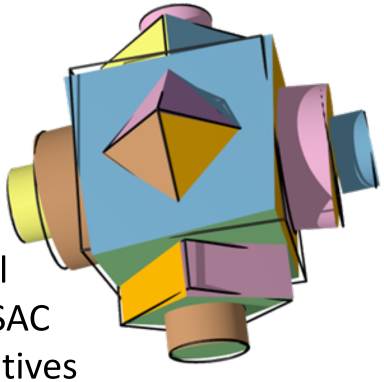
[Slide from Li et al. 11]



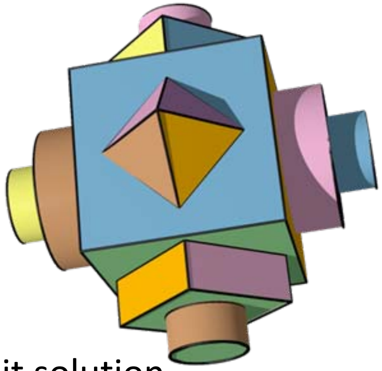
Input points



Ground truth



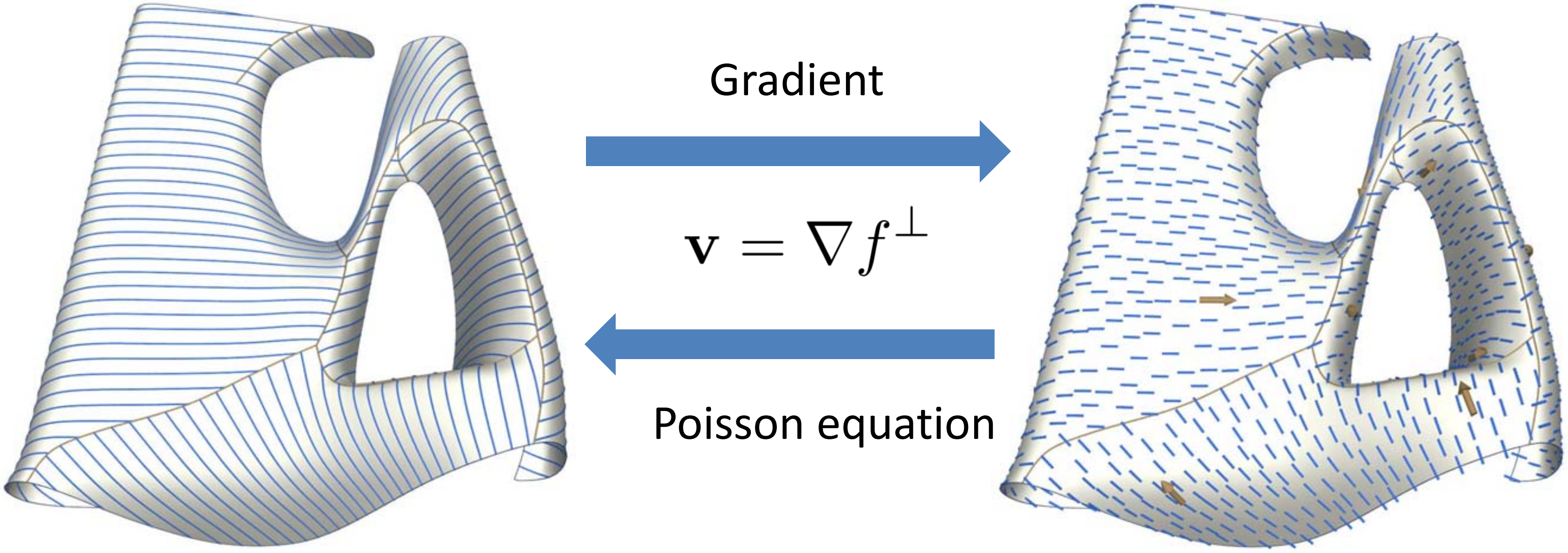
Initial RANSAC Primitives



GlobFit solution

GlobFit: Consistently Fitting Primitives by Discovering Global Relations. Yangyan Li, X. Wu, Y. Chrysanthou, A. Sharf, D. Cohen-Or and Niloy Mitra. Siggraph 2011.

Primitive Fitting in Embedded Spaces



Gradient

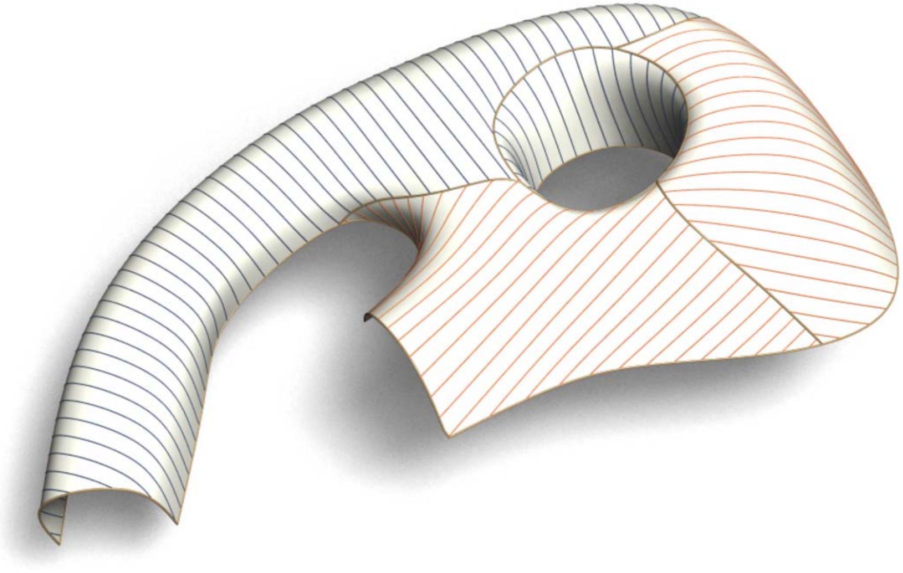
$$\mathbf{v} = \nabla f^{\perp}$$

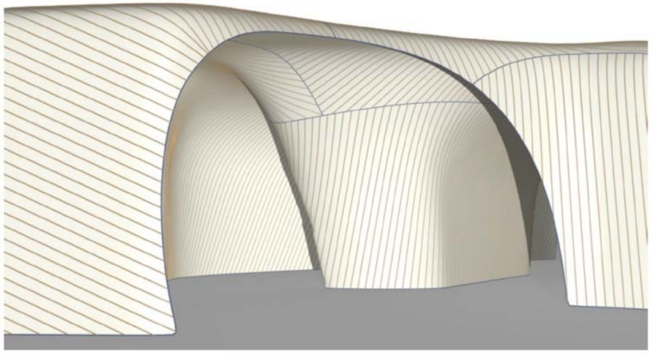
Poisson equation

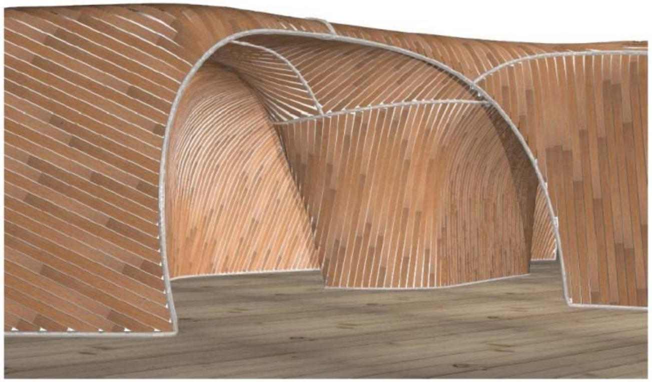
Easy to incorporate user Inputs

Helmut Pottmann, Qi-Xing Huang, Bai-Lin Deng, Alexander Schiftner, Martin Kilian, Leonidas J. Guibas, and Johannes Wallner. Geodesic patterns. ACM Trans. Graphics 29/4 (2010), #43,1-10, Proc. SIGGRAPH.

Cont-





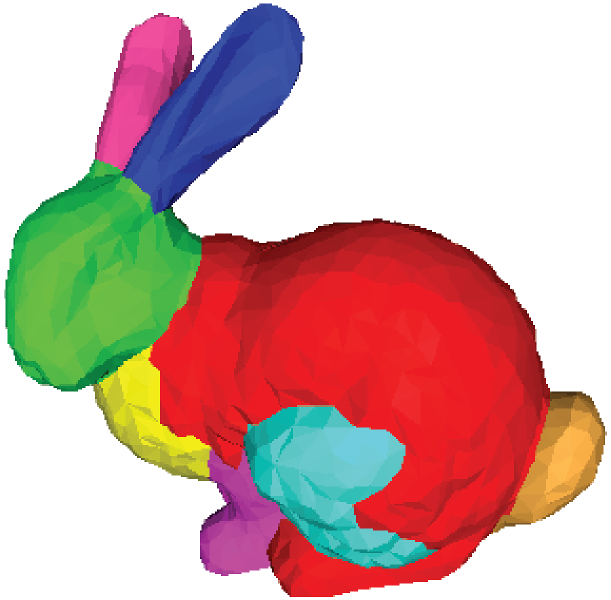


Primitive Fitting

- Based on the assumption that patches can approximately described by simple primitives
  - CAD
  - Man-made objects
- Iterative Lloyd for optimization
- Advanced primitive fitting
  - Structural constraints
  - In embedded space

Hierarchical mesh decomposition

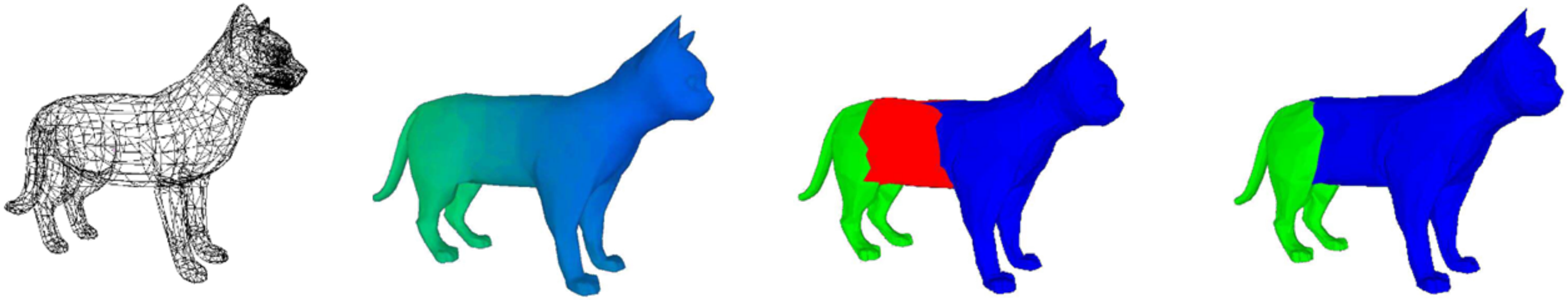
[Karz et al. 03]



Algorithm Overview (2-Way Case)

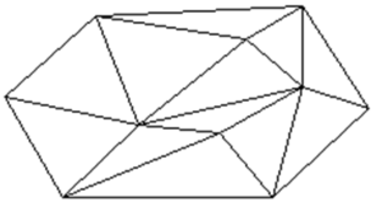
- *Criterion: faces on the same patch should be close to each other*

1. Find distances between all pairs of faces in mesh
2. Calculate probability of face belonging to each patch
3. Refine probability values using iterative clustering
4. Construct exact boundaries between components

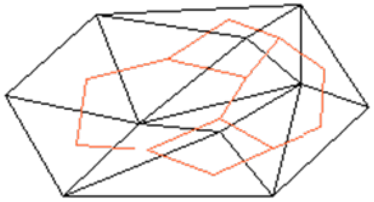


Distance Between Faces

- Shortest path along the dual graph of the input mesh



Simplex Mesh 2D

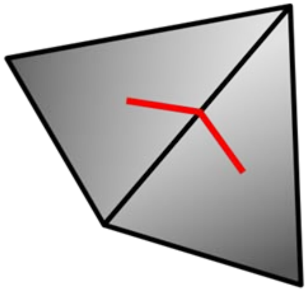


Dual Mesh

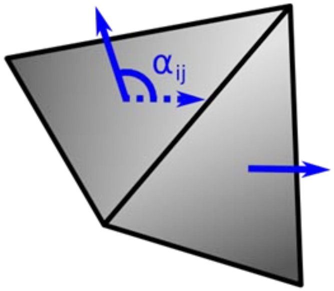
- Edge weight

$$\delta\frac{d_{geo}(f_i,f_j)}{\text{ave}(d_{geo})} + (1-\delta)\frac{d_{ang}(f_i,f_j)}{\text{ave}(d_{ang})}$$

Geodesic Distance



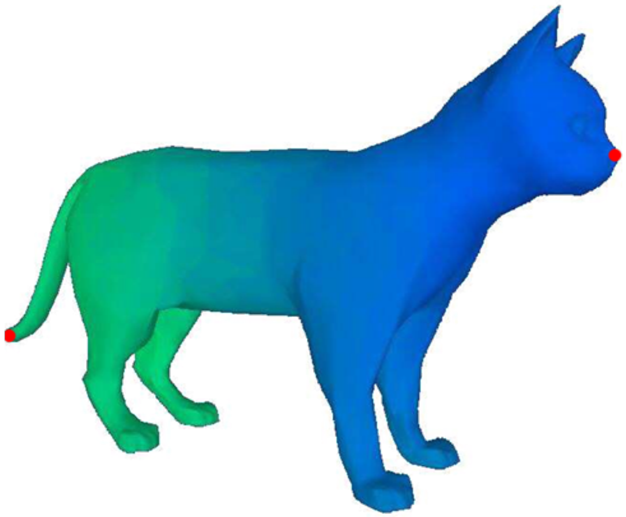
Angular Distance

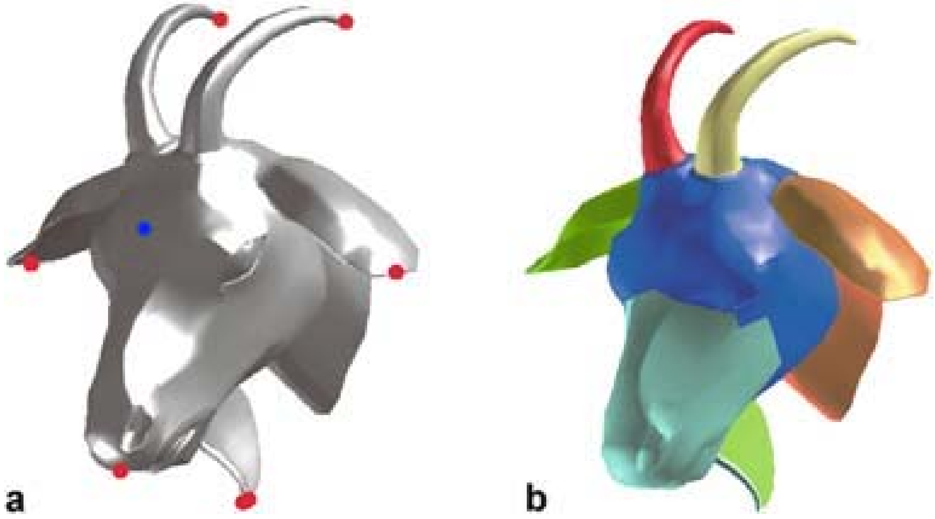


Reflects concave paths

Selecting Seed Faces

- Farthest point sampling
  - stay far away from existing seeds

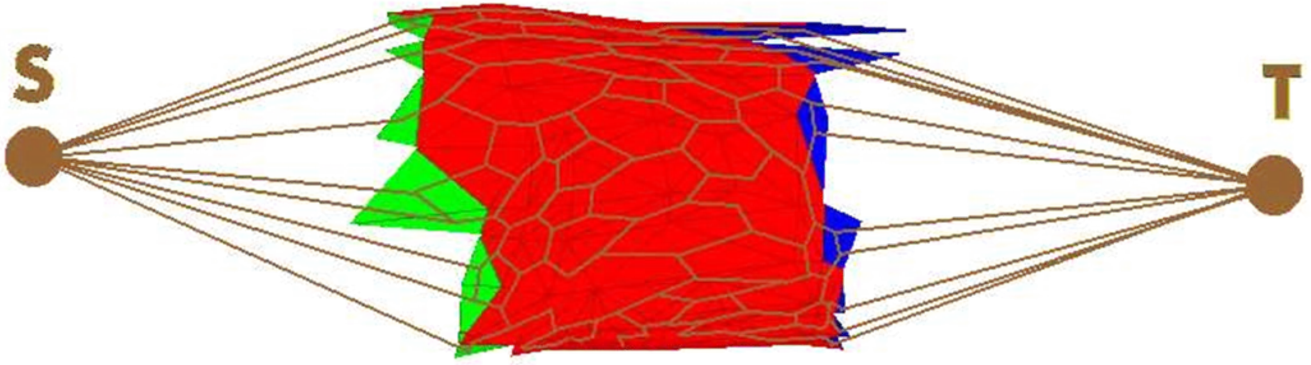




Calculate Probabilities

- Probability of face f_i belonging to patch S depends on <u>relative</u> proximity of S compared to other patches

$$\text{probability}(f_i \in S) = \frac{\text{Dist}(f_i, S_{seedface})}{\text{Dist}(f_i, S_{seedface}) + \text{Dist}(f_i, T_{seedface})}$$



Fuzzy Clustering

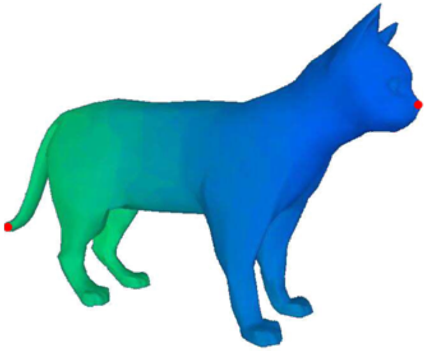
- Generating fuzzy decomposition
  - Goal: cluster faces by minimizing the function

$$F = \sum_{p} \sum_{f} \text{probability}(f \in \text{patch}(p)) \cdot \text{Dist}(f, p)$$

  - Algorithm
    - Compute the probabilities of faces belonging to each patch
    - Re-compute the seed faces to minimize F by

$$S_{seedface} = \min_{f} \sum_{f_i} \text{probability}(f_i \in S) \cdot \text{Dist}(f, f_i)$$

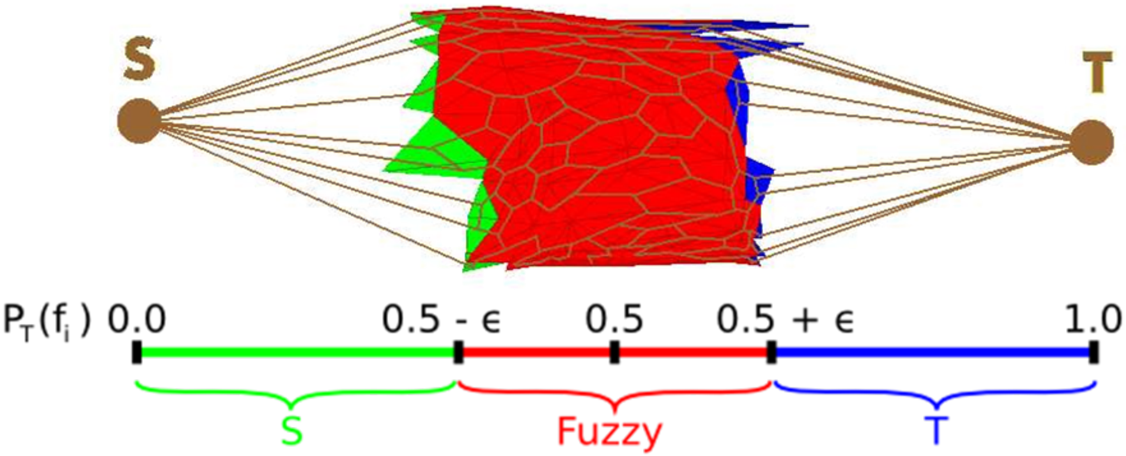
$$T_{seedface} = \min_{f} \sum_{f_i} \text{probability}(f_i \in T) \cdot \text{Dist}(f, f_i)$$



    - Iterate if the seed faces are changed

Exact Boundary

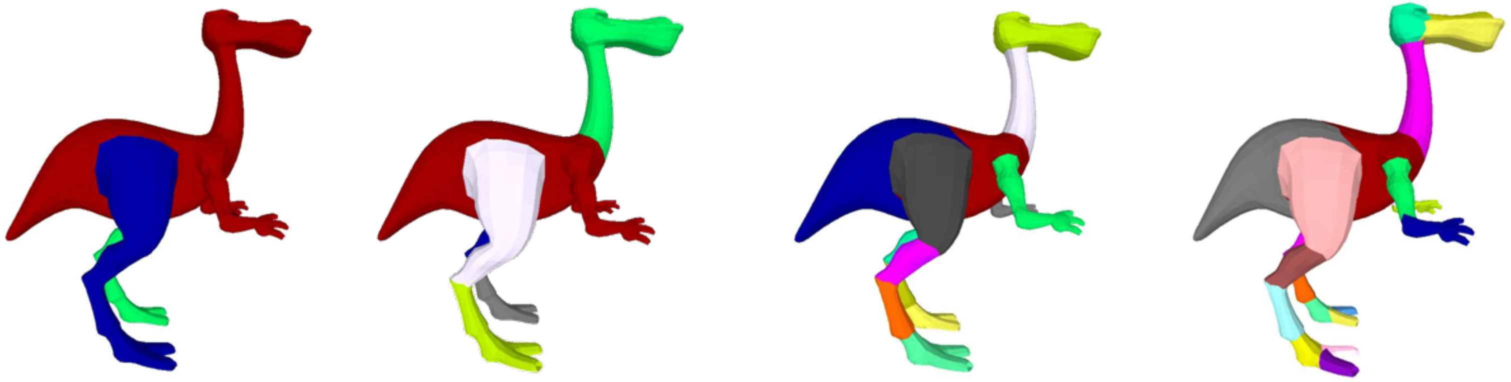
- Partition faces if probability of belonging to patch exceeds threshold (ϵ); remaining patches stay fuzzy



- Perform *min-cut* to find the boundary
  - It passes through edges with small capacities, e.g., highly concave dihedral angles.

Stopping Conditions

- Recursively decompose until either:
  - Distance between representatives < threshold
  - $\max(\alpha_{i,j}) - \min(\alpha_{i,j}) <$ threshold (faces have similar dihedral angles → patch has fairly constant curvature)
  - averageDist(Patch)/averageDist(Object) < threshold



Hierarchical Mesh Decomposition

- Represent meshes as dual graphs
- Find a meaningful graph distance metric
- Points on the same patch are close to each other
  - Fuzzy clustering
- Min-cut for extract boundaries

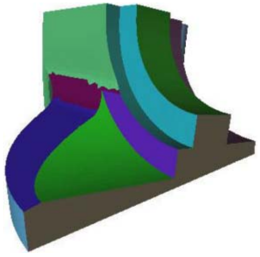
Other approaches

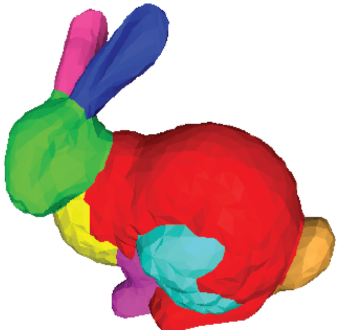
General Formulation

- Given a mesh M = {V,E,F}, find a disjoint partitioning of M into $M_1,\ldots,M_k$ such that a criterion function

$$J = J(M_1, M_2, \cdots, M_k)$$

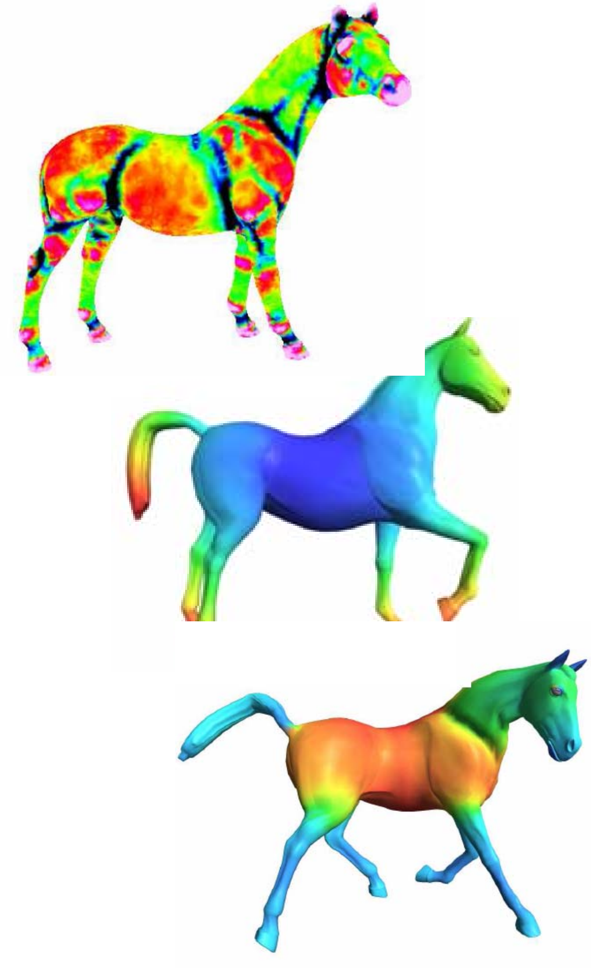
is minimized under a set of constraints C.





Types of Attributes Used

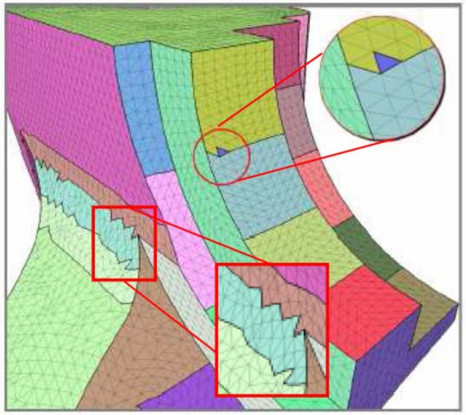
- Distance and Geodesic distance
- Planarity, normal direction
- Smoothness, curvature
- Distance to complex proxies
- Slippage
- Symmetry
- Medial Axis, Shape diameter…



Segmentation Algorithms for 3D Boundary Meshes. Ariel Shamir. Eurographics 2006 State-of-Art Report.

Types of Constraints

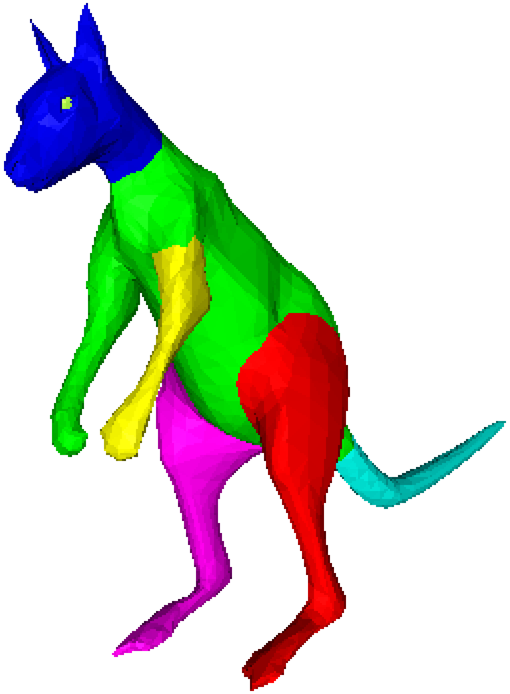
- Cardinality
  - Not too small and not too large or a given number (of segment or elements)
  - Overall balanced partition
- Geometry
  - Size: area, diameter, radius
  - Convexity, Roundness
  - Boundary smoothness
- Topology
  - Connectivity (single component)
  - Disk topology
  - a given number (of segment or elements)



Segmentation Algorithms for 3D Boundary Meshes. Ariel Shamir. Eurographics 2006 State-of-Art Report.

Randomized Cuts *[Golovinskiy and Funkhouser 08]*

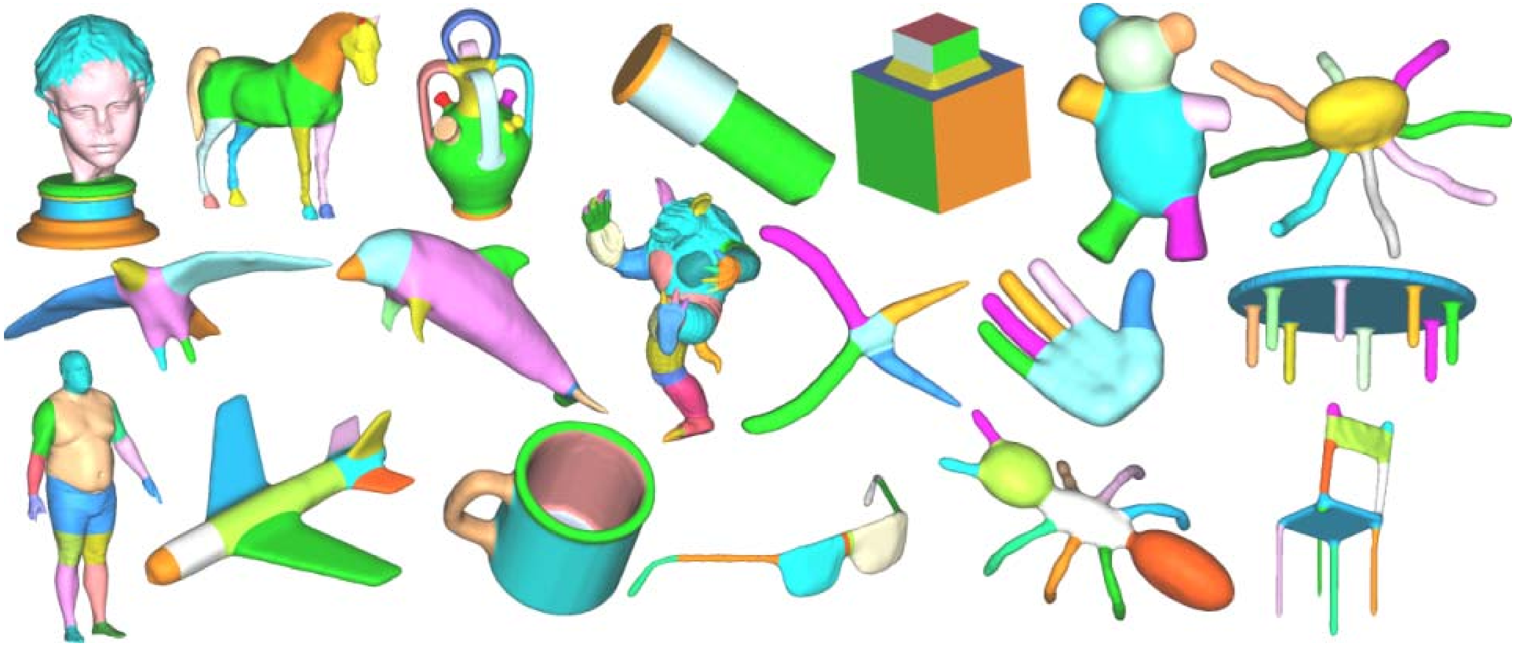
[Slide from Golovinskiy and Funkhouser 08]



Partition Function

Princeton Segmentation Benchmark [Chen et al. 09]

- 380 shapes in 19 categories
- Manual segmentations for each shape (4300 in total)



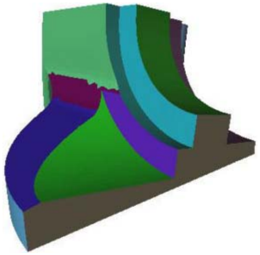
Single-Shape Segmentation

[Shalfman et al. 2002]

K-Means

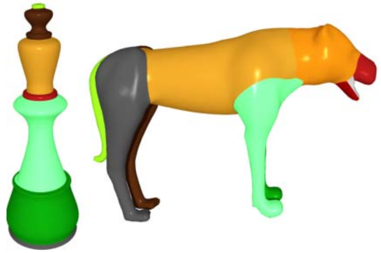
[Katz et al. 05]

Core Extraction



[Attene et. al 2006]

Fitting Primitives



[Lai et al. 08]

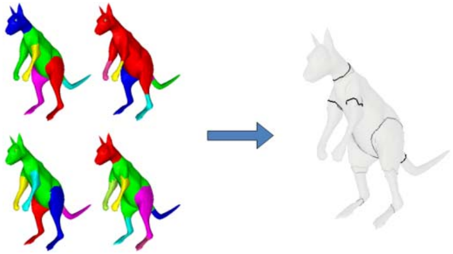
Random Walks

[Golovinskiy and Funkhouser 08]

Normalized Cuts

[Shapira et al. 08]

Shape Diameter Function



[Golovinskiy and Funkhouser 08]

Randomized Cuts

Princeton Segmentation Benchmark [Chen et al. 09]

- Evaluation metrics
  - Rand index

    The likelihood that a pair of faces are either in the same segment in two segmentations, or in different segments in both segmentations [Rand 71]

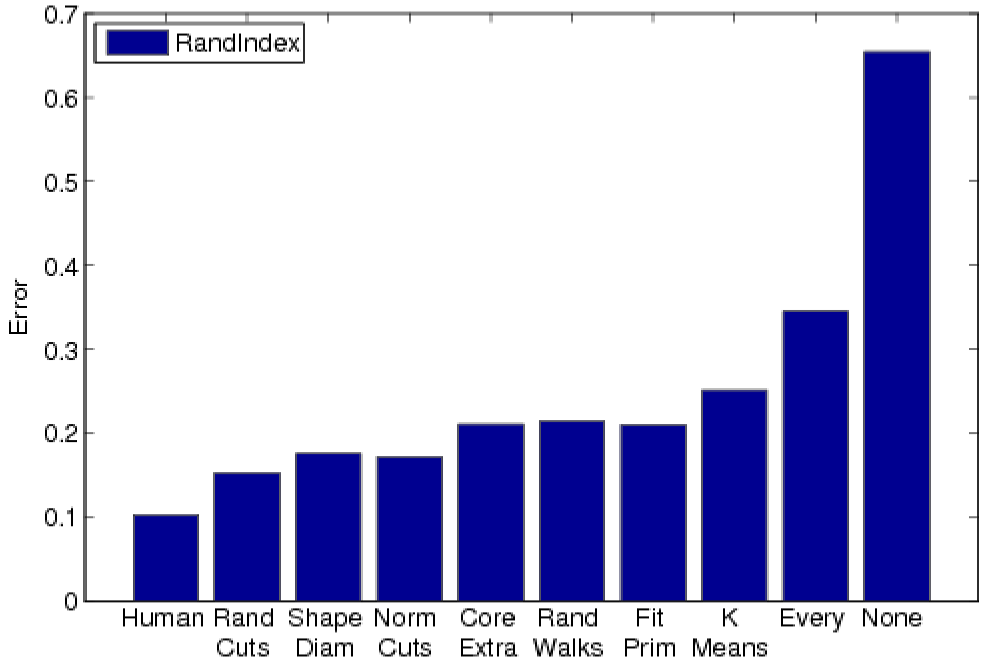
$$1-RI(S_1,S_2) = \left(\begin{array}{c}2\\N\end{array}\right)^{-1} \sum_{i,j,i<j} [C_{ij}P_{ij} + (1-C_{ij})(1-P_{ij})]$$

Same id in S_1 Same id in S_2

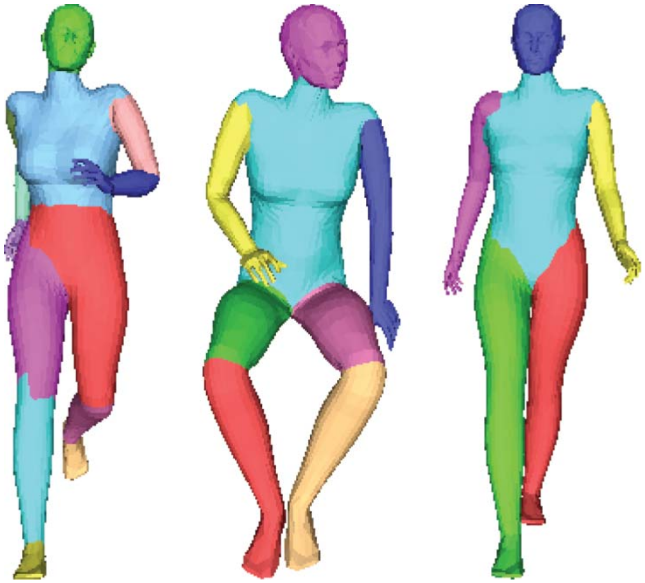
  - Averaged over all human segmentations

Princeton Segmentation Benchmark [Chen et al. 09]

- No algorithm is best for all object categories
- Human
  - Averaged rand index of all human segmentations



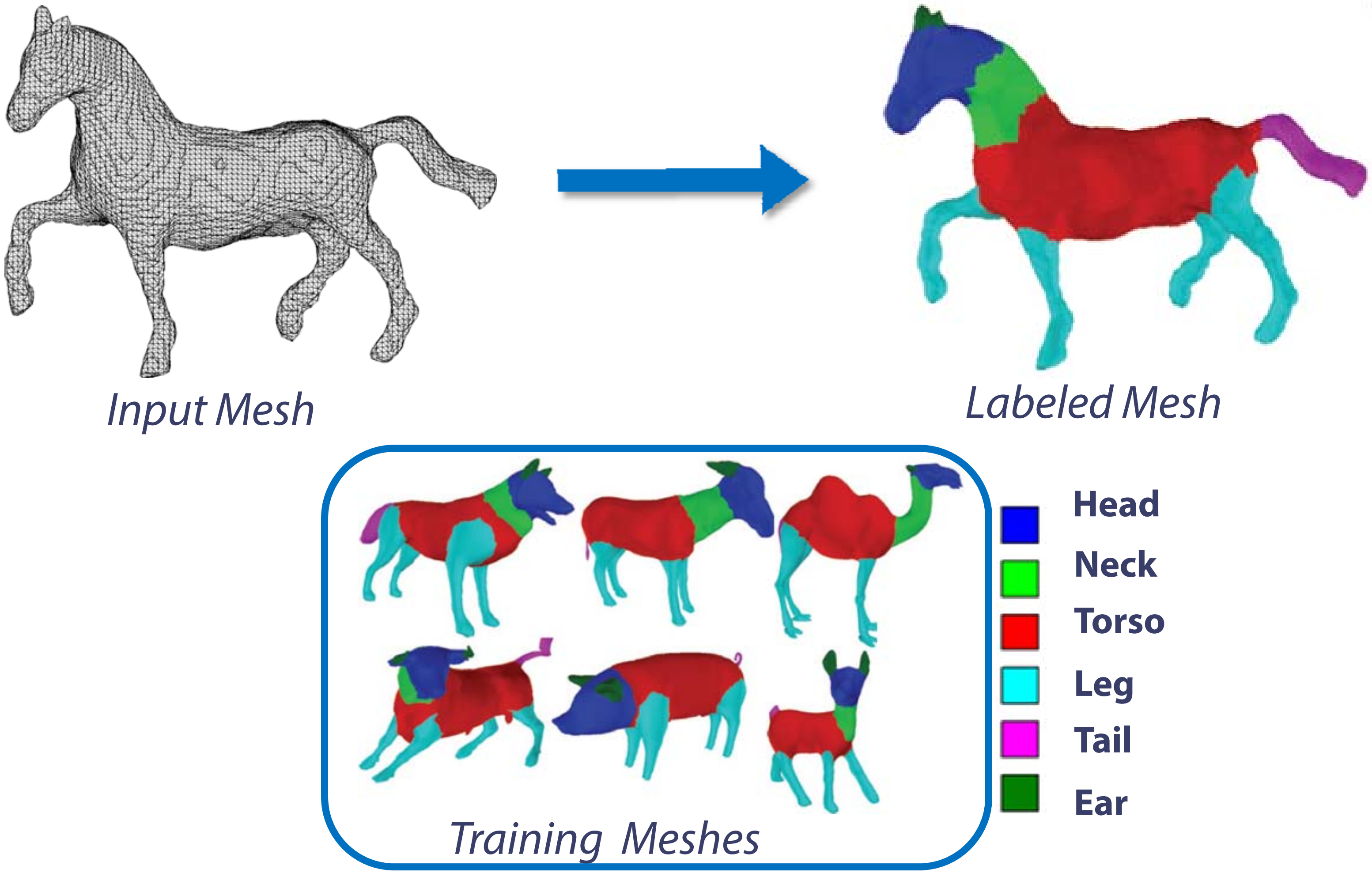
Randomized Cuts *[Golovinskiy and Funkhouser 08]*



Inconsistent across different poses

Supervised Segmentation

Goal: mesh segmentation and labeling



Input Mesh

Labeled Mesh

Training Meshes

🟦 **Head**

🟩 **Neck**

🟥 **Torso**

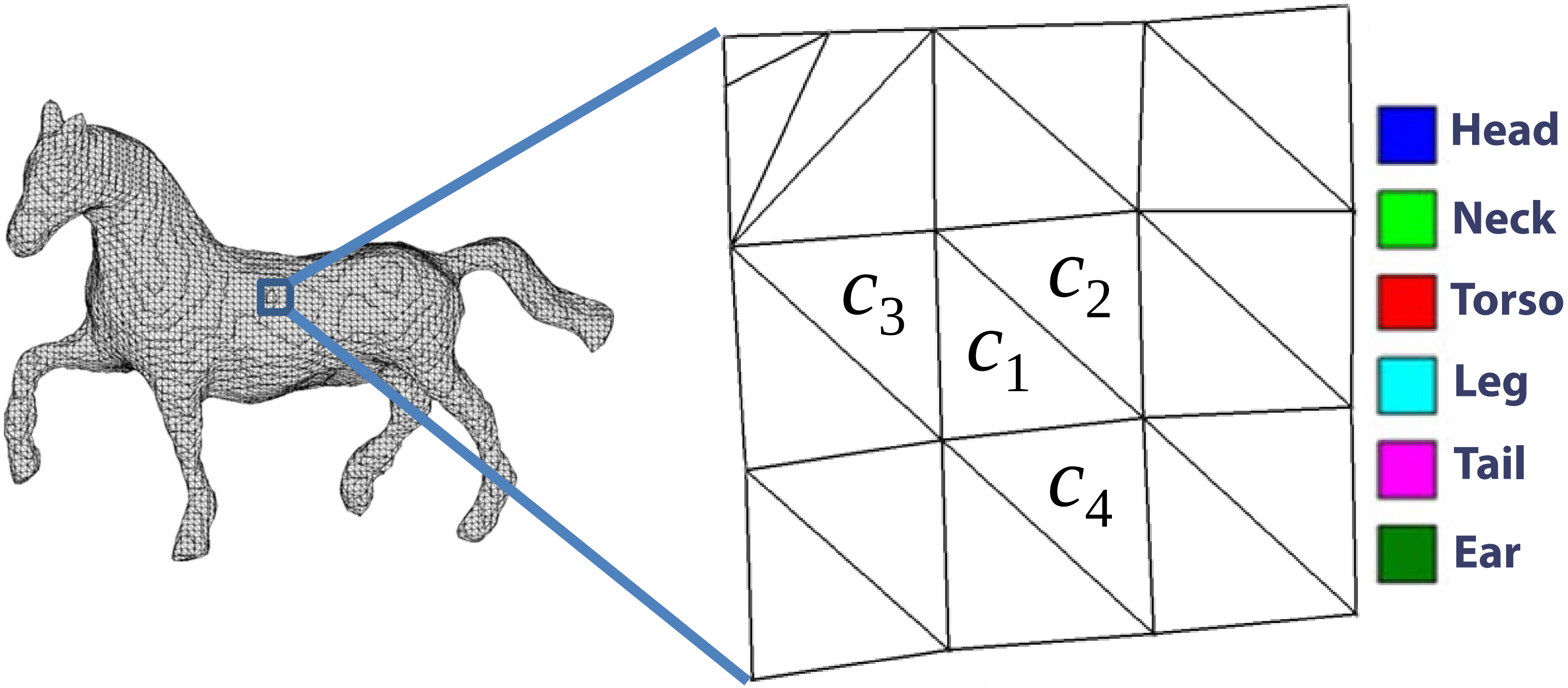
🟦 **Leg**

🟪 **Tail**

🟩 **Ear**

[Slide from Kalo et al. 11]

Labeling problem statement

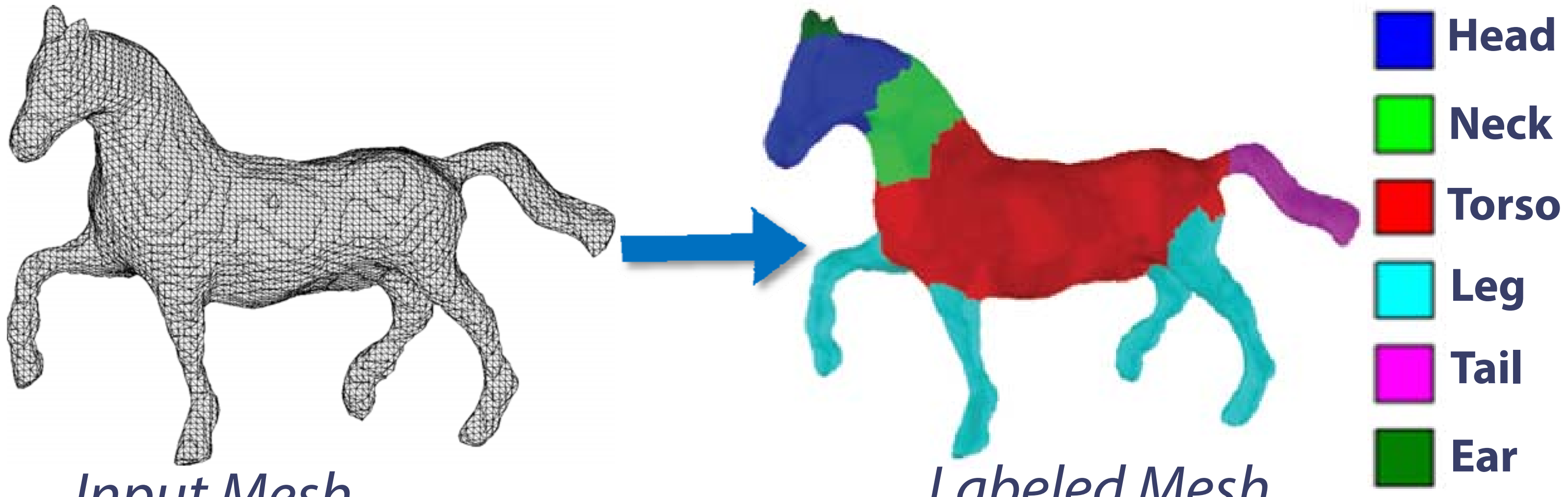


$$c_1, c_2, c_3 \in C$$

$$C = \{\ \textbf{\textit{head}},\ \textbf{\textit{neck}},\ \textbf{\textit{torso}},\ \textbf{\textit{leg}},\ \textbf{\textit{tail}},\ \textbf{\textit{ear}}\ \}$$

[Slide from Kalo et al. 11]

Conditional Random Field for Labeling



Input Mesh

Labeled Mesh

🟦 Head

🟩 Neck

🟥 Torso

🟦 Leg

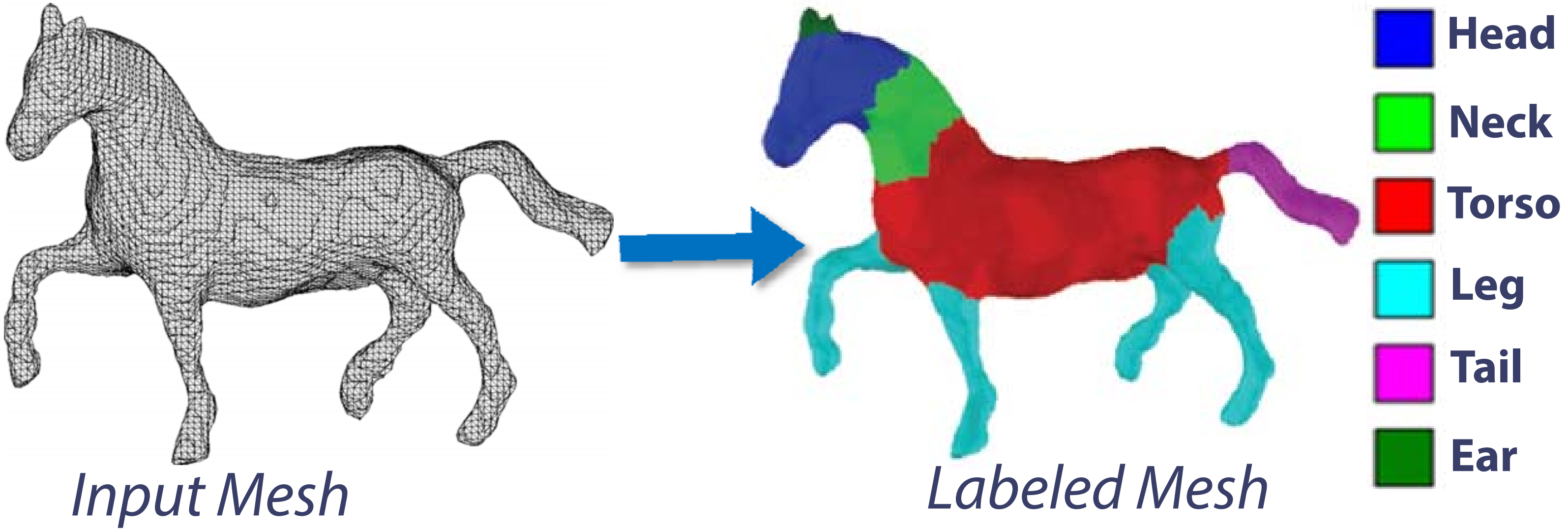
🟪 Tail

🟩 Ear

$$c^* = \arg\min_{\mathbf{c}} \left\{ \sum_{i} \alpha_i \mathrm{E}_1(c_i; \mathbf{x_i}) + \sum_{i,j} l_{ij} \mathrm{E}_2(c_i, c_j; \mathbf{y_{ij}}) \right\}$$

[Slide from Kalo et al. 11]

Conditional Random Field for Labeling

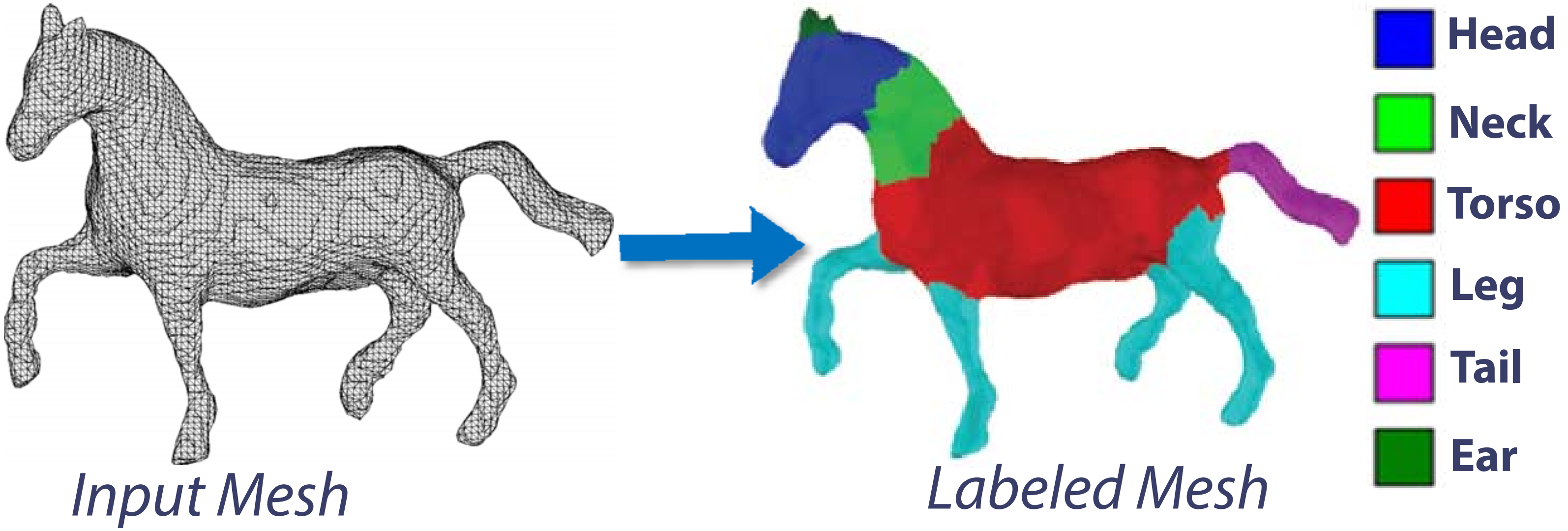


$$c^* = \arg\min_{\mathbf{c}} \left\{ \sum_{i} \alpha_i \mathrm{E}_1(c_i;\boxed{\mathbf{x_i}}) + \sum_{i,j} l_{ij} \mathrm{E}_2(c_i, c_j;\mathbf{y_{ij}}) \right\}$$

Face features

[Slide from Kalo et al. 11]

Conditional Random Field for Labeling

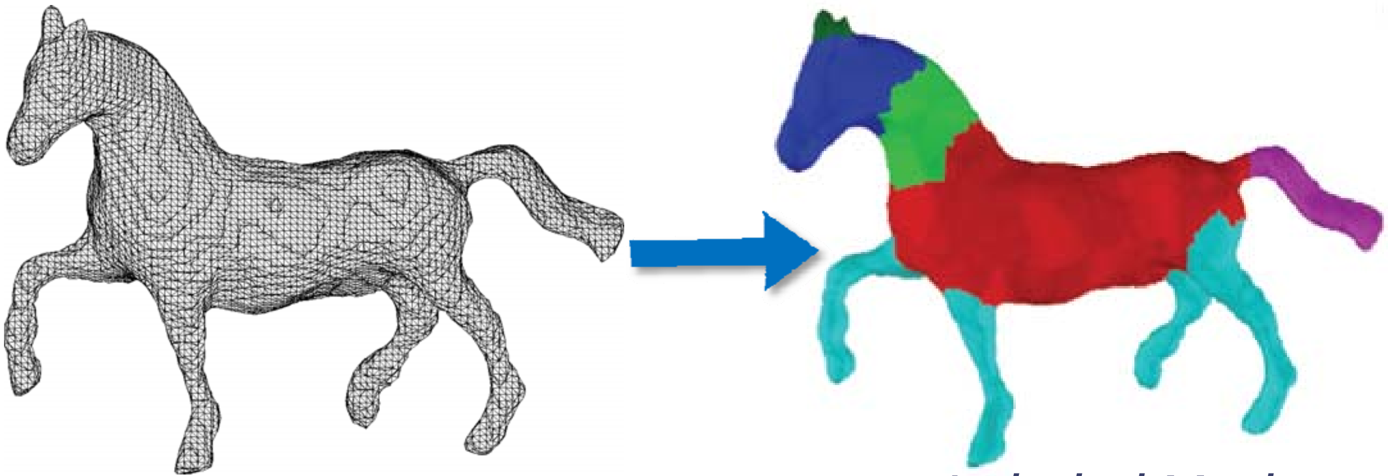


$$c^* = \arg\min_{\mathbf{c}} \left\{ \sum_{i} \boxed{\alpha_i} \mathrm{E}_1(c_i; \mathbf{x_i}) + \sum_{i,j} l_{ij} \mathrm{E}_2(c_i, c_j; \mathbf{y_{ij}}) \right\}$$

Face Area

[Slide from Kalo et al. 11]

Conditional Random Field for Labeling



Input Mesh

Labeled Mesh

🟦 Head

🟩 Neck

🟥 Torso

🟦 Leg

🟪 Tail

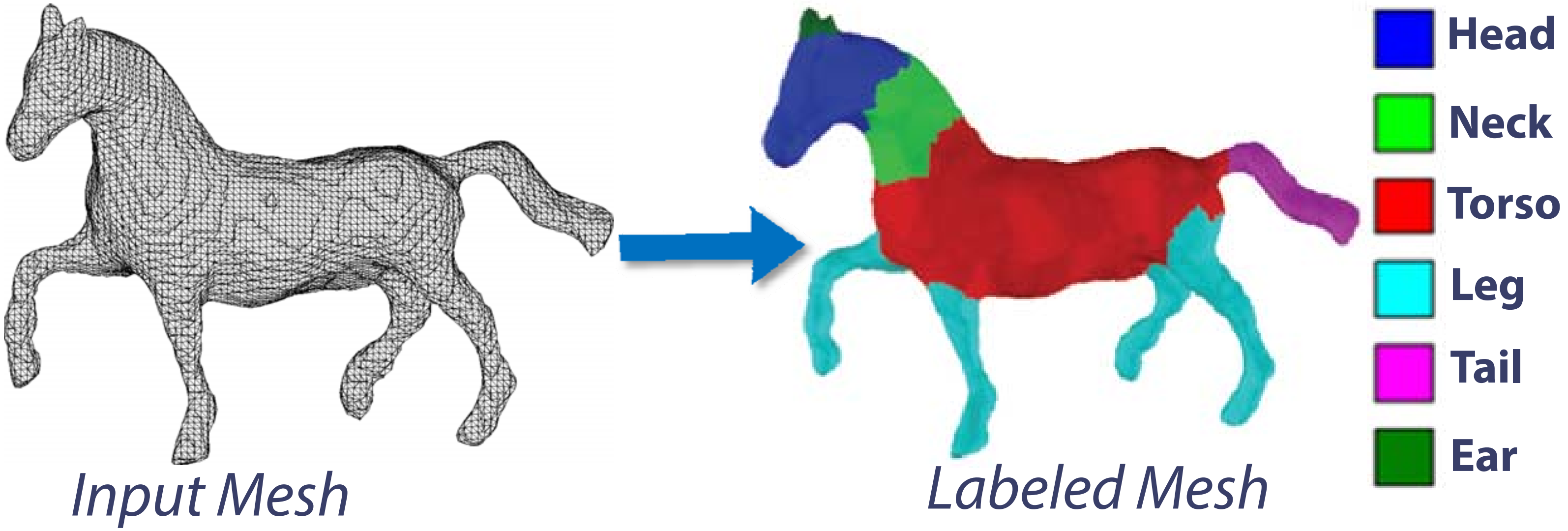
🟩 Ear

$$c^* = \arg\min_{\mathbf{c}} \left\{ \sum_{i} \alpha_i \mathrm{E}_1(c_i;\mathbf{x_i}) + \sum_{i,j} l_{ij} \boxed{\mathrm{E}_2(c_i, c_j;\mathbf{y_{ij}})} \right\}$$

Pairwise Term

[Slide from Kalo et al. 11]

Conditional Random Field for Labeling

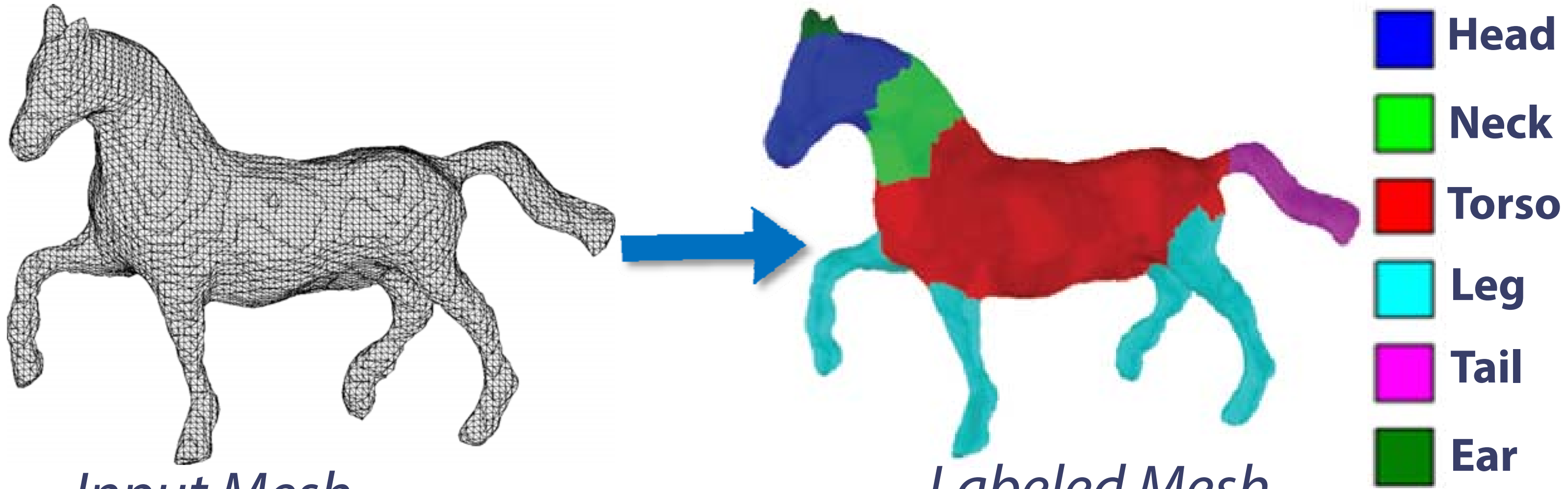


$$c^* = \arg\min_{\mathbf{c}} \left\{ \sum_{i} \alpha_i \mathrm{E}_1(c_i; \mathbf{x_i}) + \sum_{i,j} l_{ij} \mathrm{E}_2(c_i, c_j; \boxed{\mathbf{y_{ij}}}) \right\}$$

Edge Features

[Slide from Kalo et al. 11]

Conditional Random Field for Labeling



Input Mesh

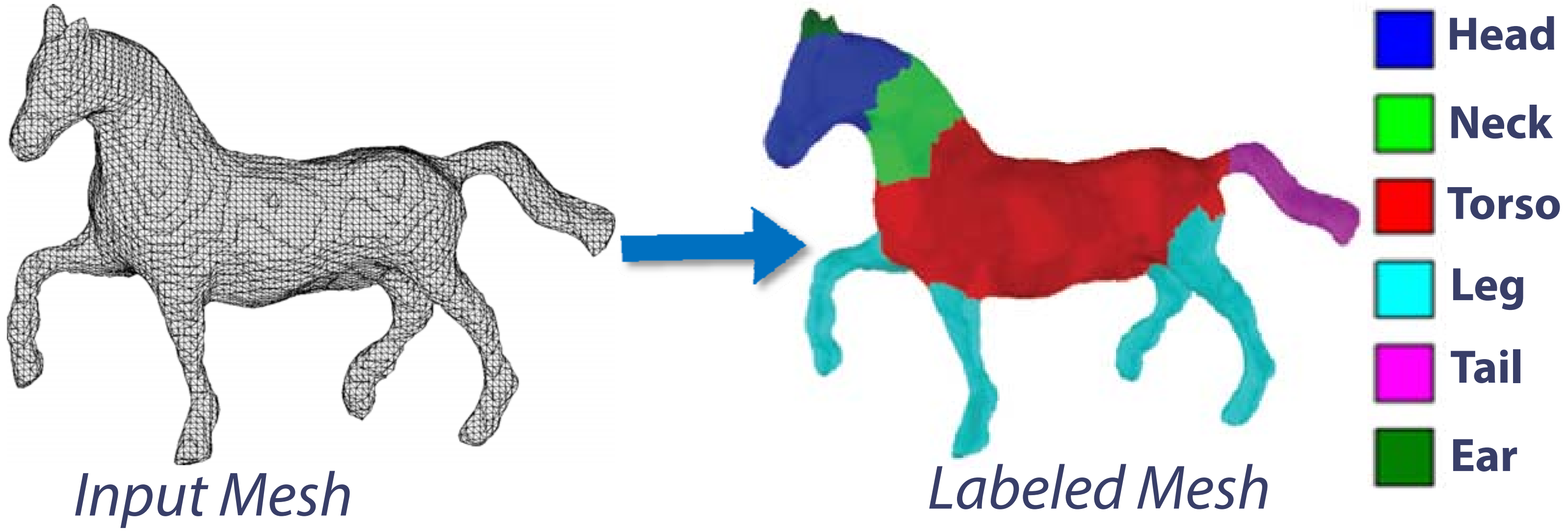
Labeled Mesh

$$c^* = \arg\min_{\mathbf{c}} \left\{ \sum_{\iota} \alpha_i \mathrm{E}_1(c_i; \mathbf{x_i}) + \sum_{\iota, j} \boxed{l_{ij}} \mathrm{E}_2(c_i, c_j; \mathbf{y_{ij}}) \right\}$$

Edge Length

[Slide from Kalo et al. 11]

Conditional Random Field for Labeling

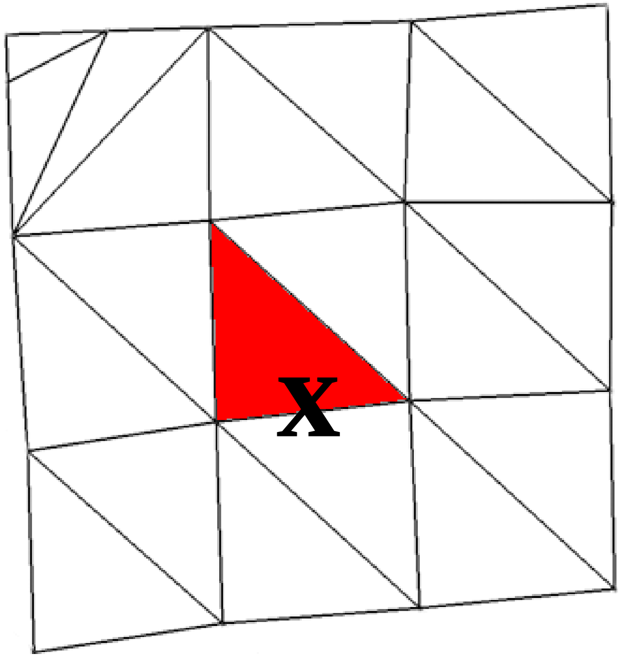


$$c^* = \arg\min_{\mathbf{c}} \left\{ \sum_{i} \alpha_i \underset{\text{Unary term}}{\boxed{E_1(c_i;\mathbf{x_i})}} + \sum_{i,j} l_{ij} E_2(c_i, c_j; \mathbf{y_{ij}}) \right\}$$

[Slide from Kalo et al. 11]

Feature vector

$$\mathbf{x} \in \Re^{375+35|C|} \rightarrow \mathrm{P}(c \mid \mathbf{x})$$

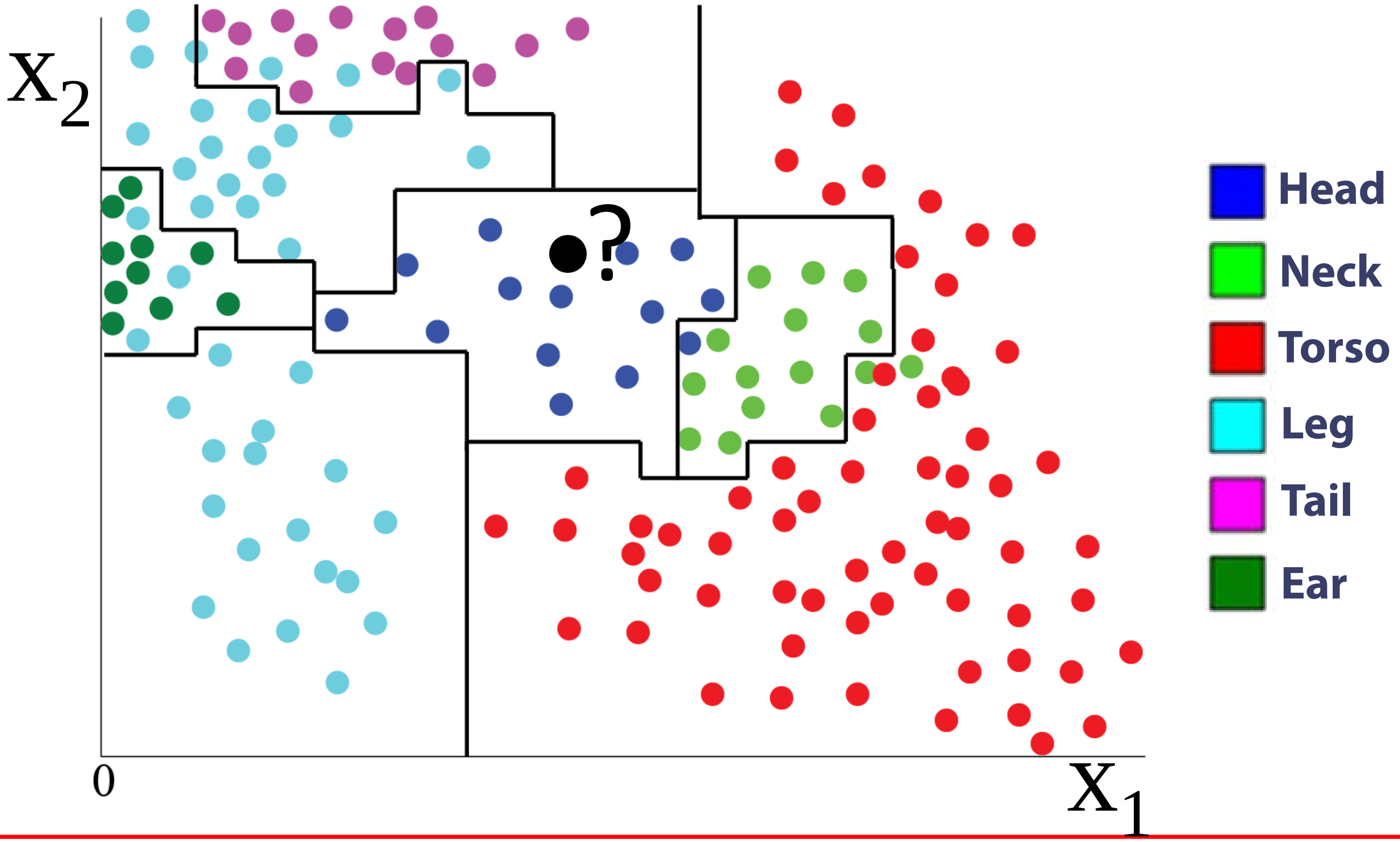


surface curvature
singular values from PCA
shape diameter
distances from medial surface
average geodesic distances
shape contexts
spin images
contextual label features
Use more features help

[Slide from Kalo et al. 11]

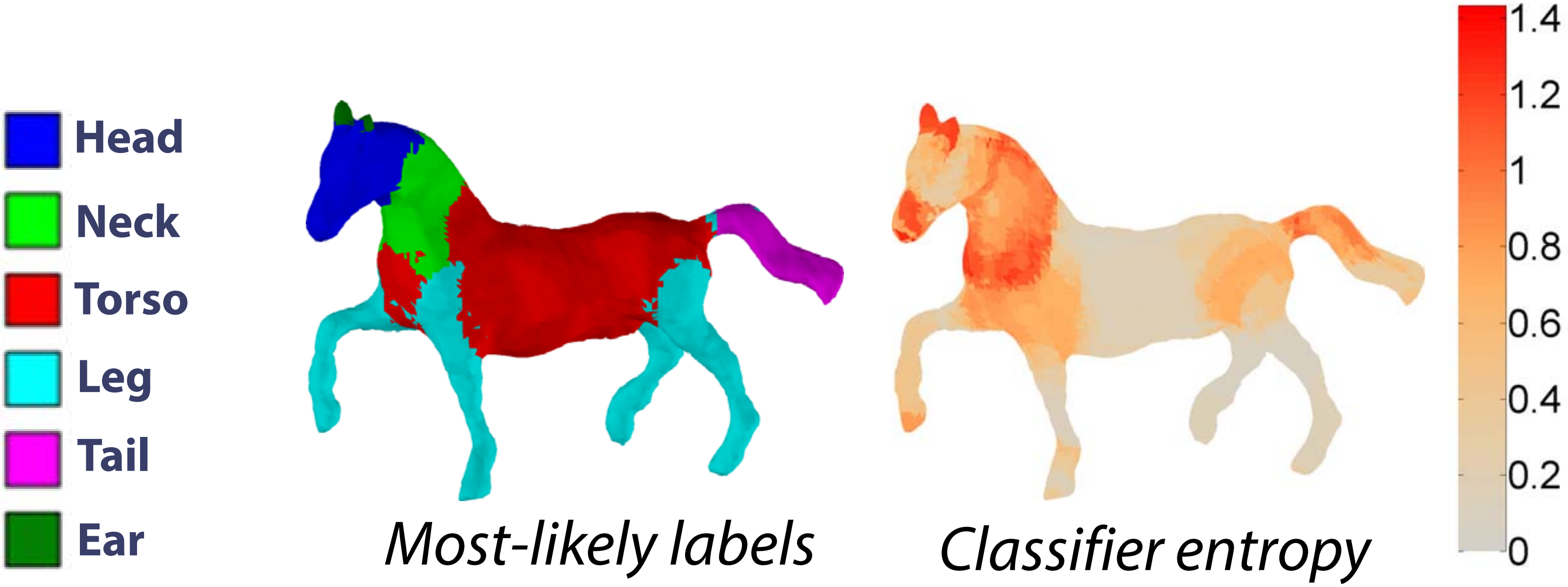
Learning a classifier

Jointboost classifier **[Torralba et al. 2007]**



[Slide from Kalo et al. 11]

Unary Term

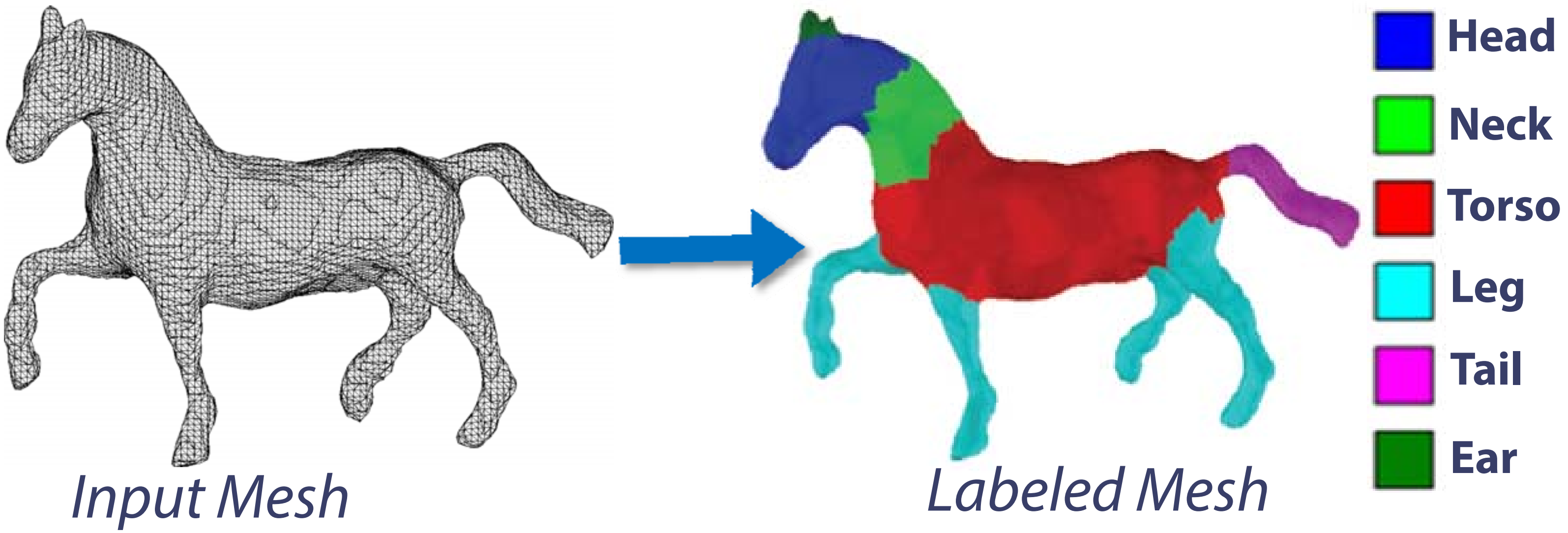


Most-likely labels

Classifier entropy

[Slide from Kalo et al. 11]

Conditional Random Field for Labeling



$$c^* = \arg\min_{\mathbf{c}} \left\{ \sum_{i} \alpha_i \mathrm{E}_1(c_i;\mathbf{x_i}) + \sum_{i,j} l_{ij} \boxed{\mathrm{E}_2(c_i,c_j;\mathbf{y_{ij}})} \right\}$$

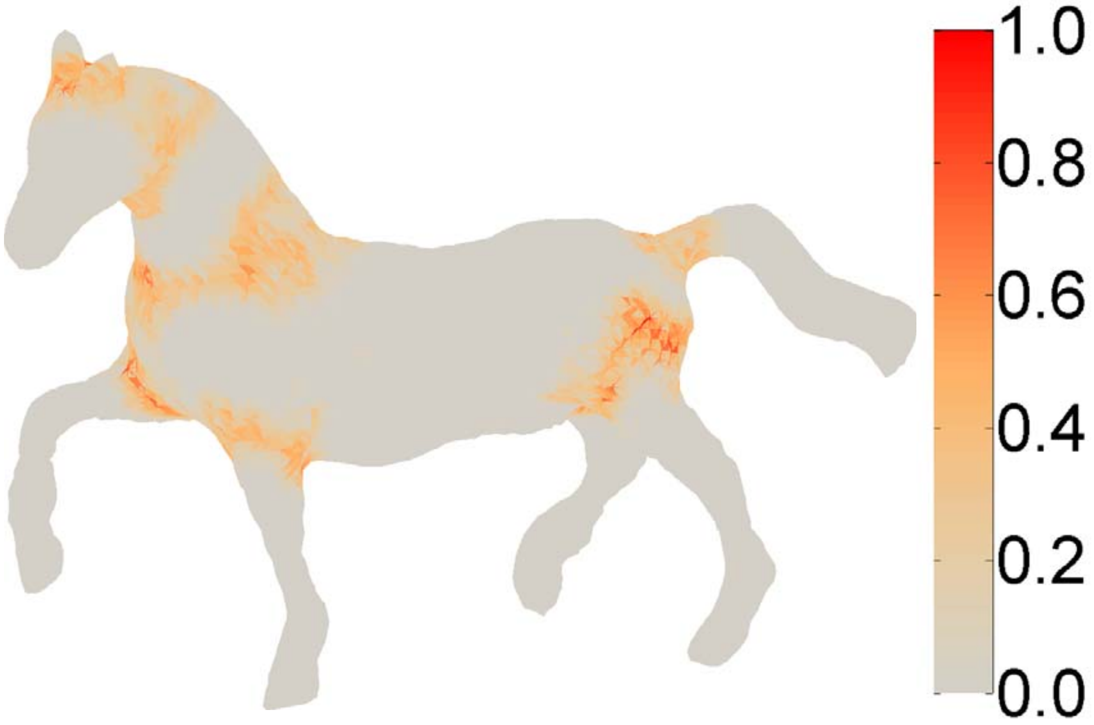
Pairwise Term

[Slide from Kalo et al. 11]

Pairwise Term

$$E_2(c,c';\mathbf{y},\theta_2) = \boxed{G(\mathbf{y})} L(c,c')$$

Geometry-dependent term



[Slide from Kalo et al. 11]

Pairwise Term

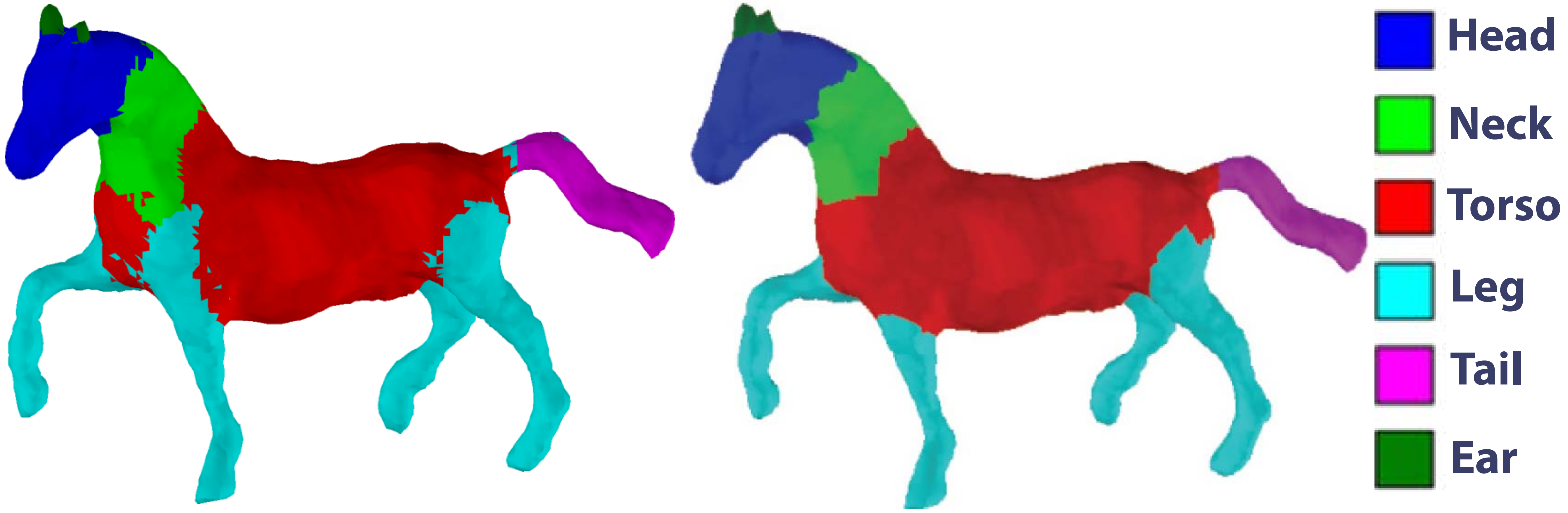
$$\mathrm{E}_2(c,c';\mathbf{y},\theta_2) = G(\mathbf{y})\boxed{L(c,c')}$$

Label compatibility term

$$L(c,c') = \begin{array}{cccccc c}
\text{Head} & \text{Neck} & \text{Ear} & \text{Torso} & \text{Leg} & \text{Tail} & \\
0 & .45 & .07 & 1 & \infty & \infty & \text{Head} \\
.45 & 0 & \infty & 1 & \infty & \infty & \text{Neck} \\
.07 & \infty & 0 & \infty & \infty & \infty & \text{Ear} \\
1 & 1 & \infty & 0 & 1 & .56 & \text{Torso} \\
\infty & \infty & \infty & 1 & 0 & \infty & \text{Leg} \\
\infty & \infty & \infty & .56 & \infty & 0 & \text{Tail}
\end{array}$$

[Slide from Kalo et al. 11]

Full CRF result

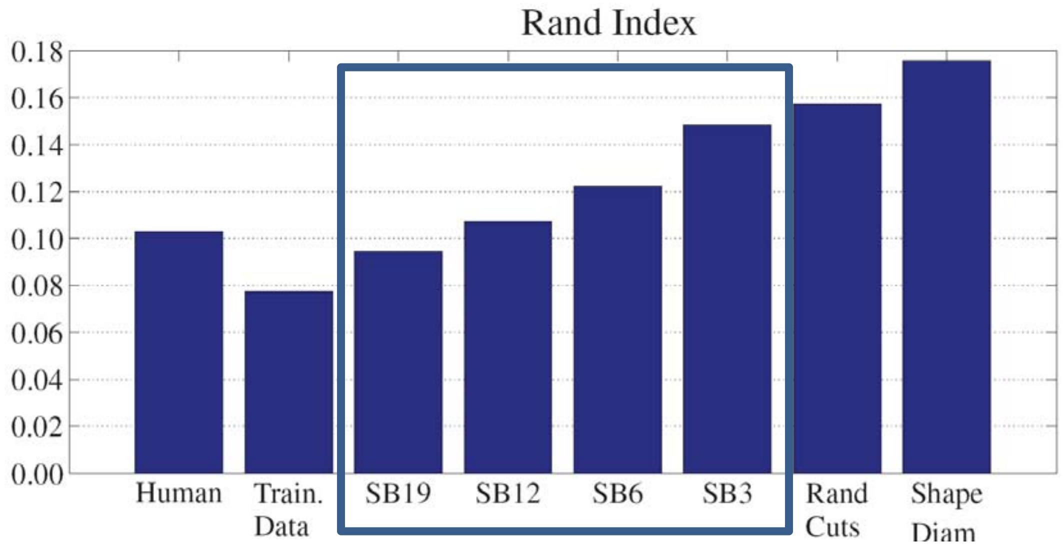


Unary term classifier *Full CRF result*

[Slide from Kalo et al. 11]

Supervised Segmentation [Kalogerakis et al.10]

- Significant improvements from single-shape segmentations



- Limitations
  - Prior knowledge of the category
  - Shape variation within each category shall be small

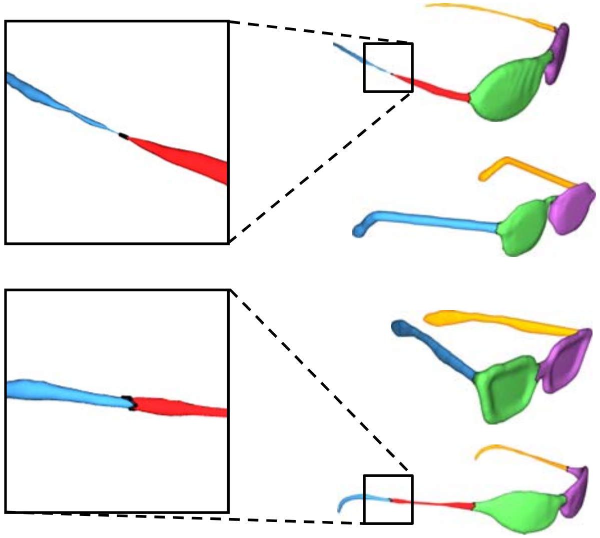
Joint Shape Segmentation

Motivations

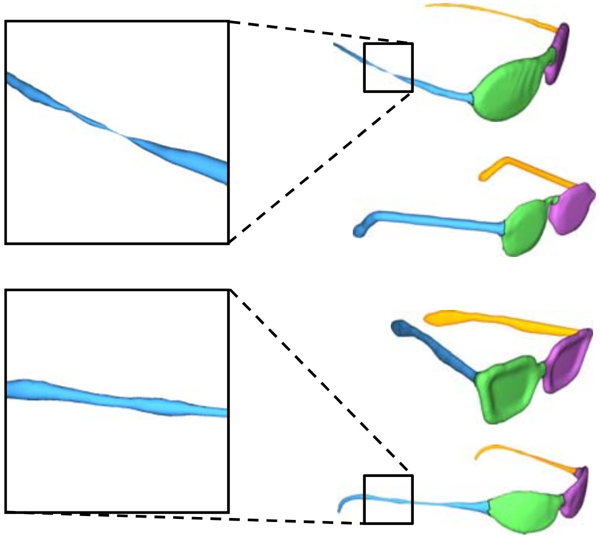
Structural similarity of segmentations

- Extraneous geometric clues

Single shape segmentation
[Chen et al. 09]



Joint shape segmentation
[Huang et al. 11]

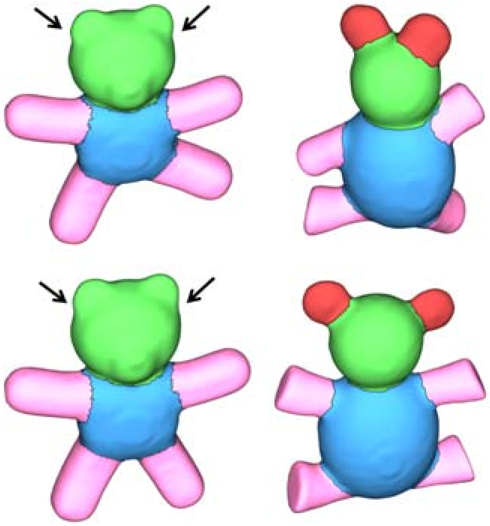


Motivations

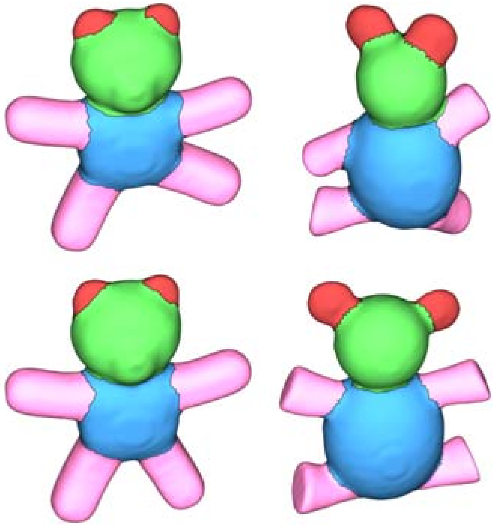
Structural similarity of segmentations

- Low saliency

Single shape segmentation
[Chen et al. 09]



Joint shape segmentation
[Huang et al. 11]

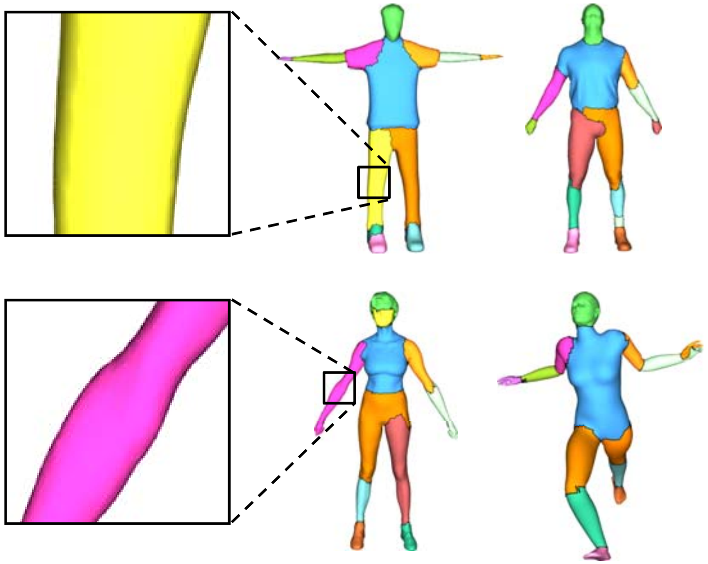


Motivations

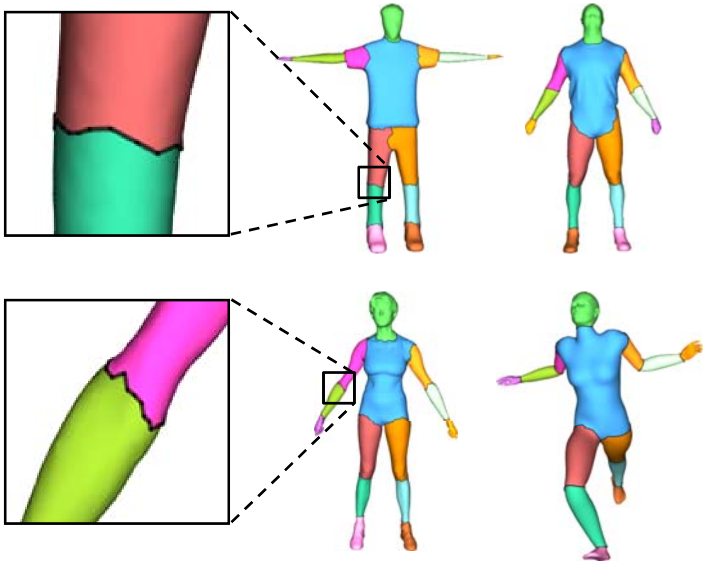
(Rigid) invariance of segments

- Articulated structures

Single shape segmentation
[Chen et al. 09]



Joint shape segmentation
[Huang et al. 11]

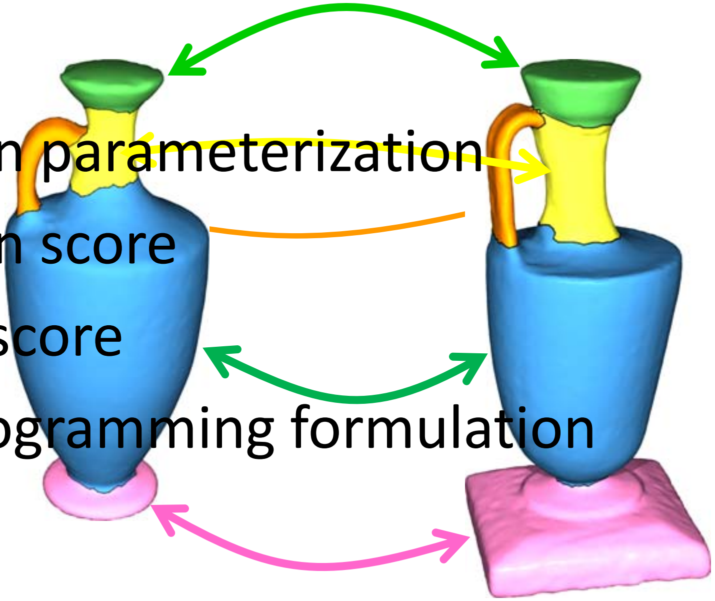


Pair-wise Joint Segmentation

Objective:

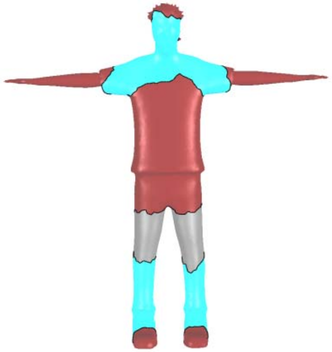
$$\max_{S_1,S_2} \text{score}(S_1)+\text{score}(S_2)+\text{consistency}(S_1,S_2)$$

Outline:

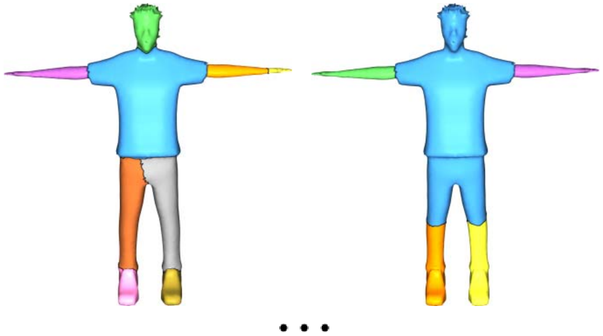


- Segmentation parameterization
- Segmentation score
- Consistency score
- 0-1 linear programming formulation

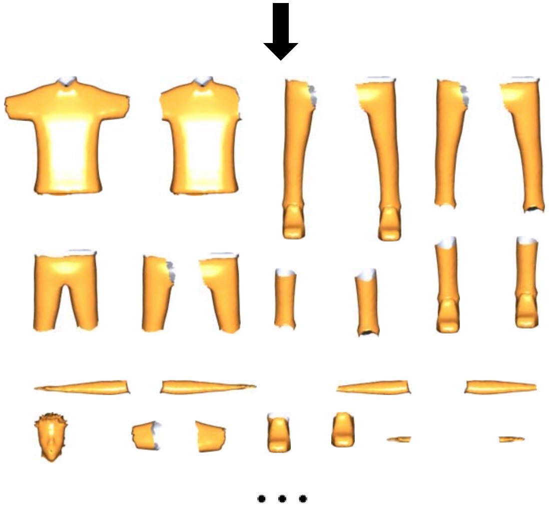
Segmentation Parameterization



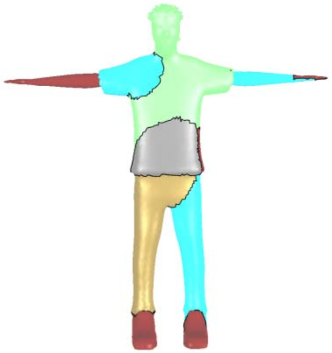
Shape Diameter
[Shapira et al. 08]



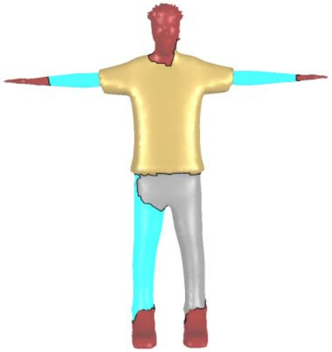
Randomized Cuts



Initial Segments



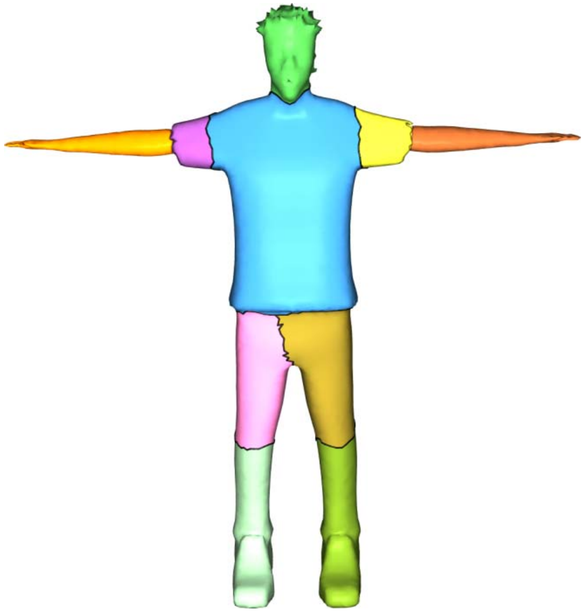
Random Walks
[Lai et al. 08]

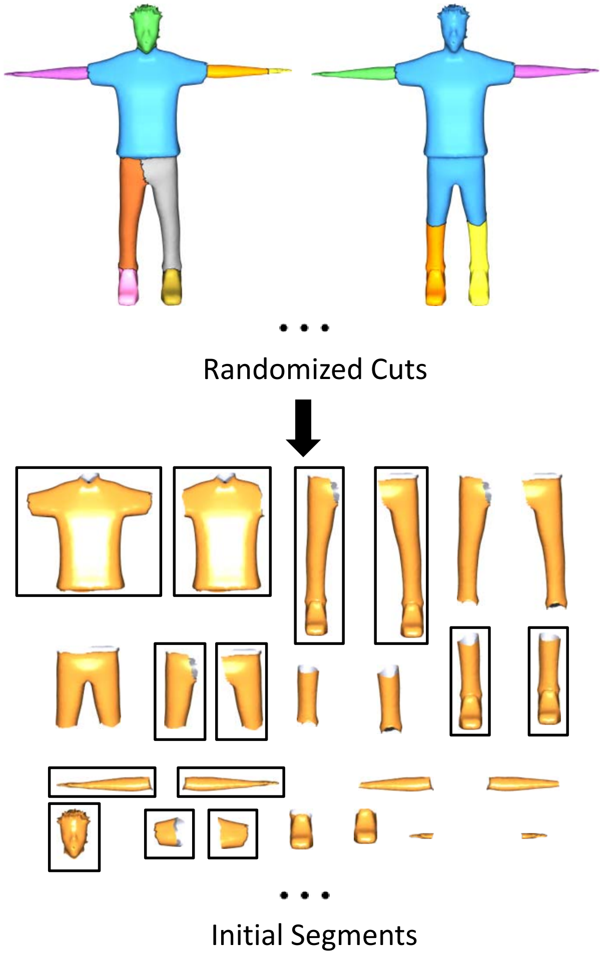


Normalized Cuts
[Golovinskiy and Funkhouser 08]

Segmentation Parameterization

- Segmentations: subsets of initial segments obtained from randomized segmentations



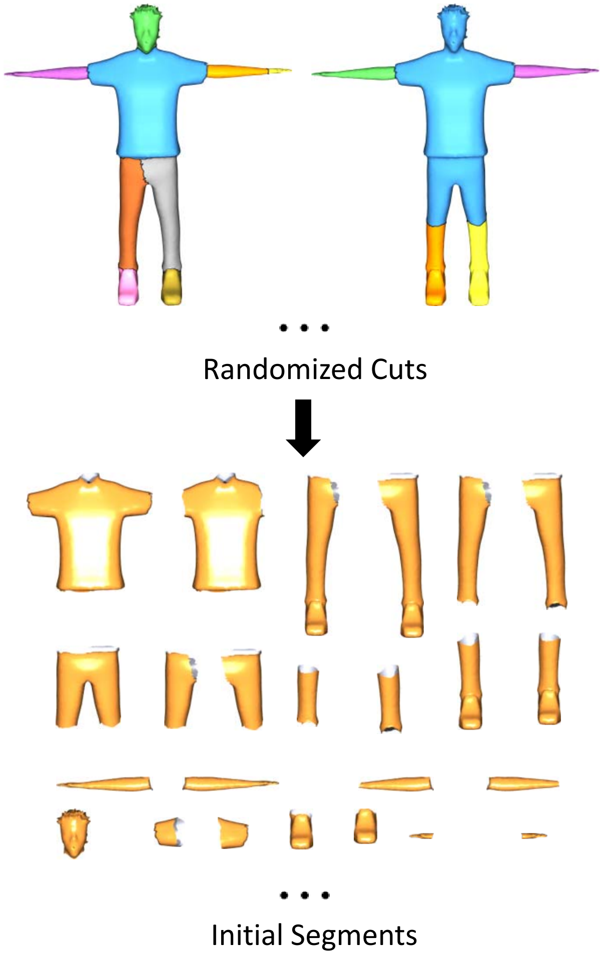


Segmentation Parameterization

- Segmentations: subsets of initial segments obtained from randomized segmentations
- Segmentation constraints: each point is in exactly one segment

$$|\text{cover}(p)| = 1, \quad \forall p \in W$$

The set of initial segments that cover point p



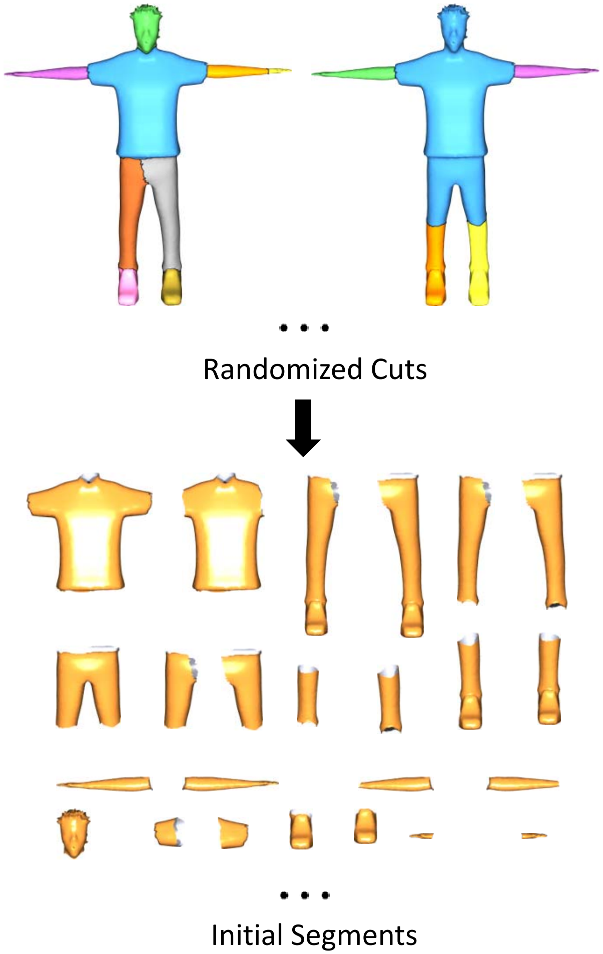
Segmentation Parameterization

- Segmentations: subsets of initial segments obtained from randomized segmentations
- Segmentation constraints: each point is in exactly one segment
- Segmentation score

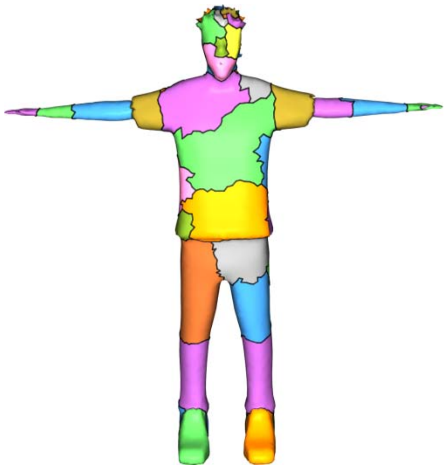
$$\text{score}(S) = \sum_{s\in S} \overline{\text{area}}(s)\, r_s = \sum_{s\in S} \overline{w}_s$$

Prevent tiny segments

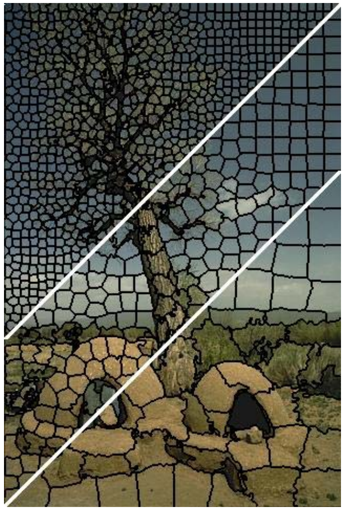
Repetitions



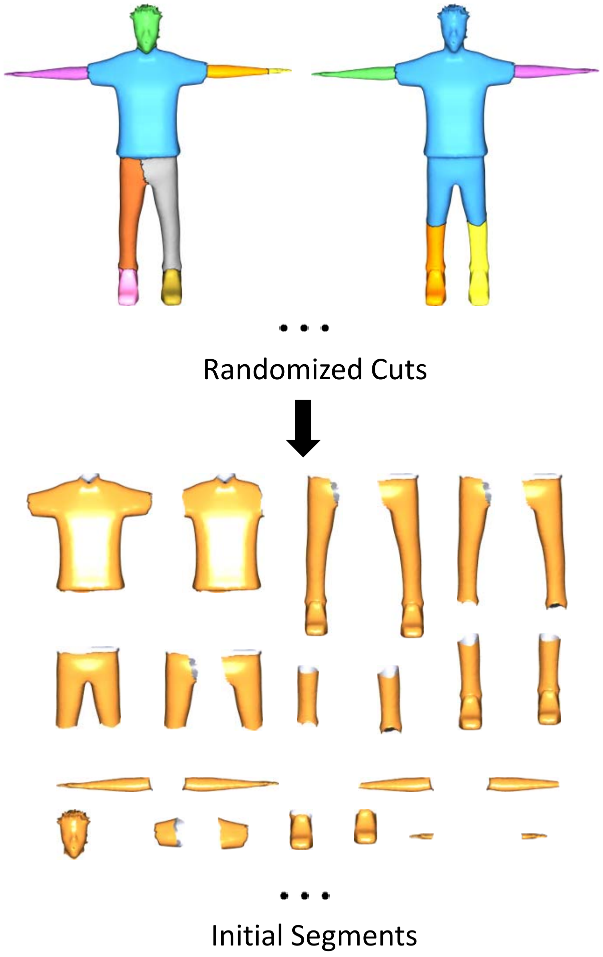
Segmentation Parameterization



Patches
[Golovinskiy and Funkhouser 08]

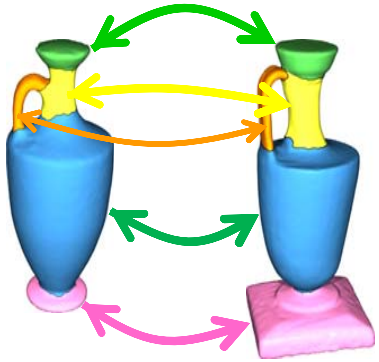


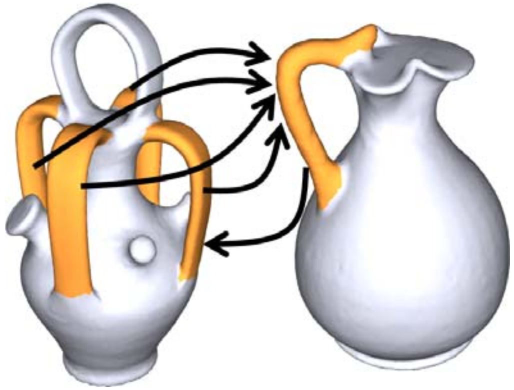
Super-pixels
[Ren and Malik 03]



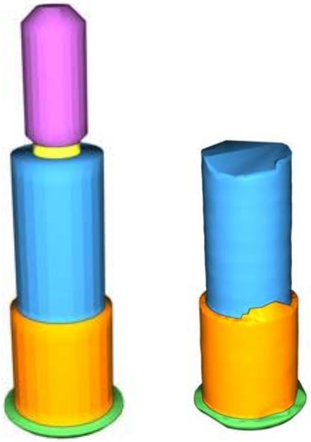
Consistency Term

- Defined in terms of mappings
  - Oriented
  - Partial





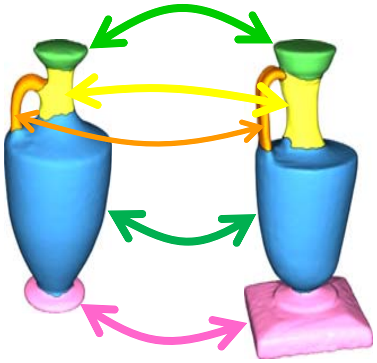
Many-to-one correspondences



Partial similarity

Consistency Term

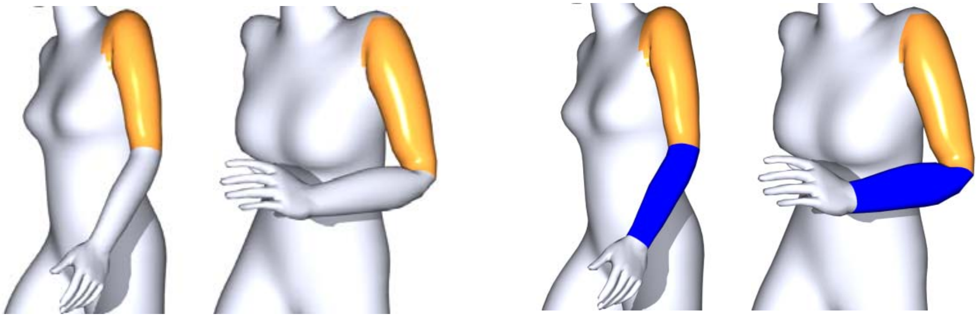
- Defined in terms of mappings
  - Oriented
  - Partial
- Mapping score [Anguelov et al.05]



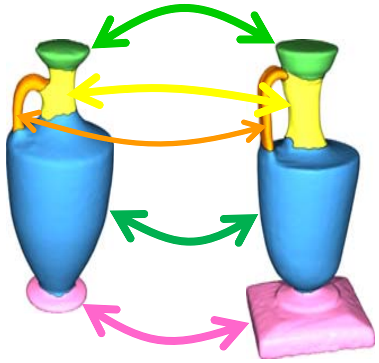
$$\mathrm{score}(\mathcal{M}_{ij}) = \lambda \boxed{\sum_{c\in\mathcal{M}_{ij}} \overline{w}_c} + \mu \boxed{\sum_{(c,c')\in\mathcal{A}_{ij}} \overline{w}_{(c,c')}}$$

Correspondence weight [Osoda et al. 02]

Adjacent correspondence pair weight



Consistency Term



- Defined in terms of mappings
  - Oriented
  - Partial
- Mapping score [Anguelov et al.05]

$$\text{score}(\mathcal{M}_{ij}) = \lambda \sum_{c \in \mathcal{M}_{ij}} \overline{w}_c + \mu \sum_{(c,c') \in \mathcal{A}_{ij}} \overline{w}_{(c,c')}$$

- Consistency score

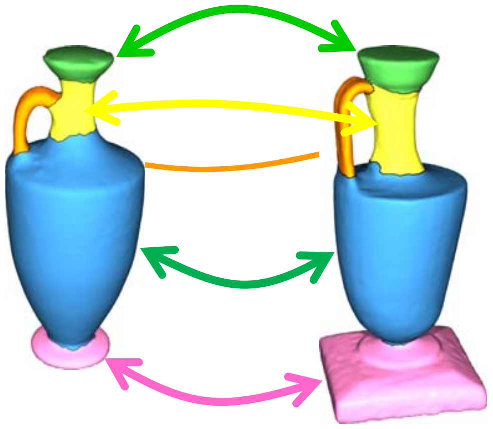
$$\text{consistency}(S_1, S_2) = \sum_{ij \in \{12,21\}} \max_{\mathcal{M}_{ij}} \text{score}(\mathcal{M}_{ij})$$

Constrained Optimization

$$\max_{S_1,S_2,\mathcal{M}_{12},\mathcal{M}_{21}} \sum_{i=1}^{2} \sum_{s\in S_i} \overline{w}_s + \sum_{ij\in\{12,21\}} (\lambda \sum_{c\in\mathcal{M}_{ij}} \overline{w}_c + \mu \sum_{(c,c')\in\mathcal{A}_{ij}} \overline{w}_{(c,c')})$$

$$\text{s.t.} \qquad |\text{cover}(p)| = 1, \quad \forall p \in \mathcal{P}_i, \quad 1 \leq i \leq 2,$$

$$\mathcal{M}_{ij} \in \text{Mapping}(\mathcal{S}_i \times \mathcal{S}_j), \quad ij \in \{12, 21\}$$

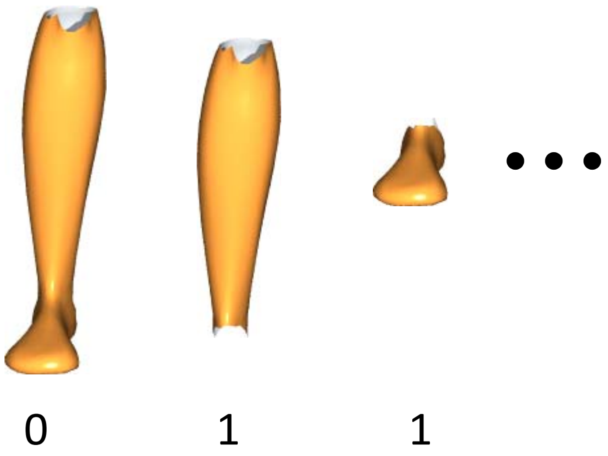


0-1 Linear Programming Formulation

- Introduce binary indicators

Segments

$$x_s = \begin{cases} 1 & s \in S_1 \cup S_2 \\ 0 & \text{otherwise} \end{cases}$$



0-1 Linear Programming Formulation

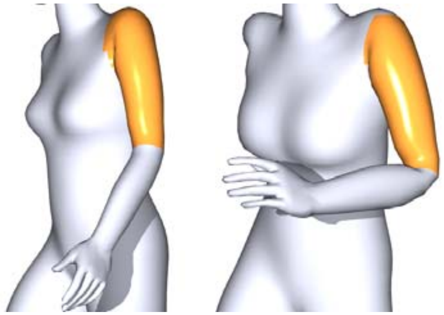
- Introduce binary indicators

Segments

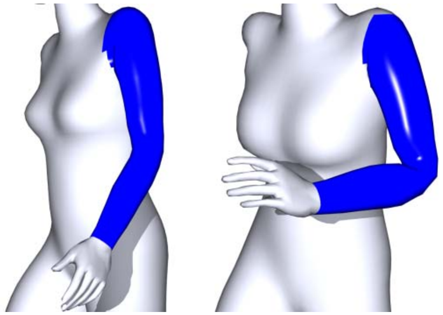
$$x_s = \begin{cases} 1 & s \in S_1 \cup S_2 \\ 0 & \text{otherwise} \end{cases}$$

Correspondences

$$y_c = \begin{cases} 1 & c \in \mathcal{M}_{12} \cup \mathcal{M}_{21} \\ 0 & \text{otherwise} \end{cases}$$



1



0

…

0-1 Linear Programming Formulation

- Introduce binary indicators

Segments

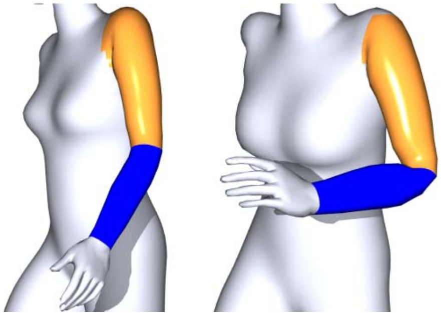
$$x_s = \begin{cases} 1 & s \in S_1 \cup S_2 \\ 0 & \text{otherwise} \end{cases}$$

Correspondences

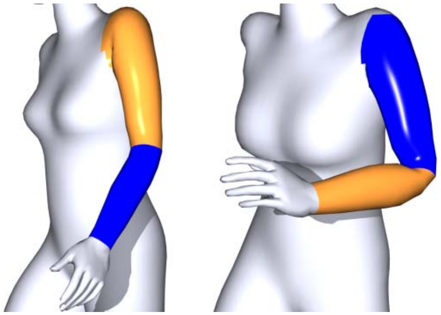
$$y_c = \begin{cases} 1 & c \in \mathcal{M}_{12} \cup \mathcal{M}_{21} \\ 0 & \text{otherwise} \end{cases}$$

Correspondence pairs

$$z_{(c,c')} = \begin{cases} 1 & (c, c') \in \mathcal{A}_{12} \cup \mathcal{A}_{21} \\ 0 & \text{otherwise} \end{cases}$$



1



0

• • •

0-1 Linear Programming Formulation

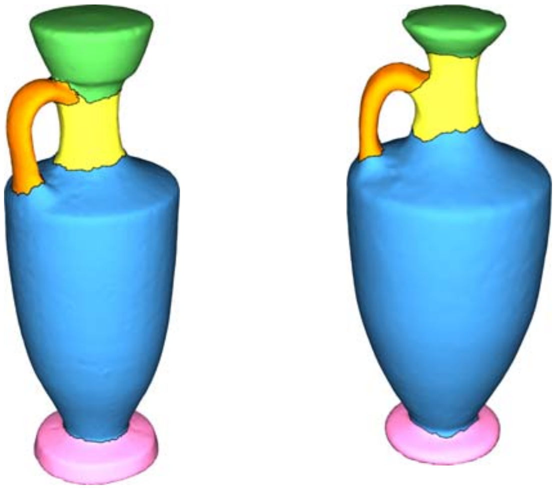
Linear programming relaxation

$$\max \sum_{i\in\{1,2\}} \mathbf{x}_i^{\mathsf{T}}\mathbf{w}_i^{\mathsf{seg}} + \sum_{ij\in\{12,21\}} (\lambda \mathbf{y}_{ij}^{\mathsf{T}}\mathbf{w}_{ij}^{\mathsf{corr}} + \mu \mathbf{z}_{ij}^{\mathsf{T}}\mathbf{w}_{ij}^{\mathsf{adj}})$$

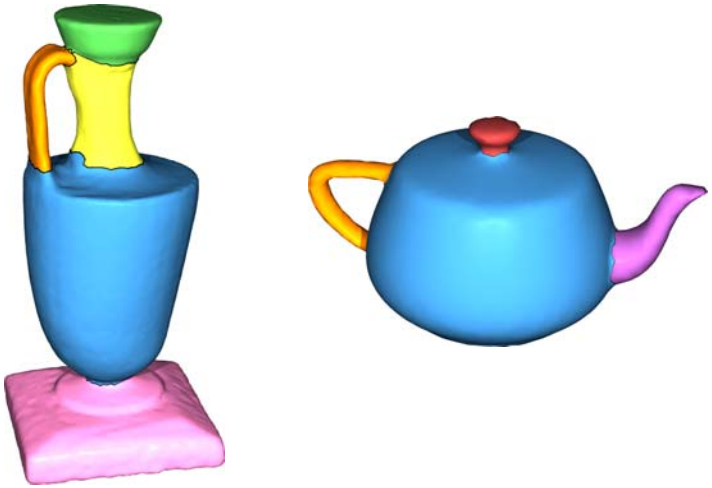
$$\begin{array}{lll}
\text{s.t.} & A_1\mathbf{x}_1 = 1 & A_2\mathbf{x}_2 = 1 \\
& B_{12}\mathbf{y}_{12} \le D_{12}\mathbf{x}_1 & B_{21}\mathbf{y}_{21} \le D_{21}\mathbf{x}_2 \\
& B_{12}^{'}\mathbf{y}_{12} \le D_{12}^{'}\mathbf{x}_2 & B_{21}^{'}\mathbf{y}_{21} \le D_{21}^{'}\mathbf{x}_1 \\
& E_{12}\mathbf{z}_{12} \le F_{12}\mathbf{y}_{12} & E_{21}\mathbf{z}_{21} \le F_{21}\mathbf{y}_{21} \\
\text{and} & 0 \le x \le 1 & \forall x \in \mathbf{x}_1, \mathbf{x}_2, \mathbf{y}_{12}, \mathbf{y}_{21}, \mathbf{z}_{12}, \mathbf{z}_{21}
\end{array}$$

Similar Shapes

- As a by-product, pair-wise joint segmentation determines pairs of similar shapes



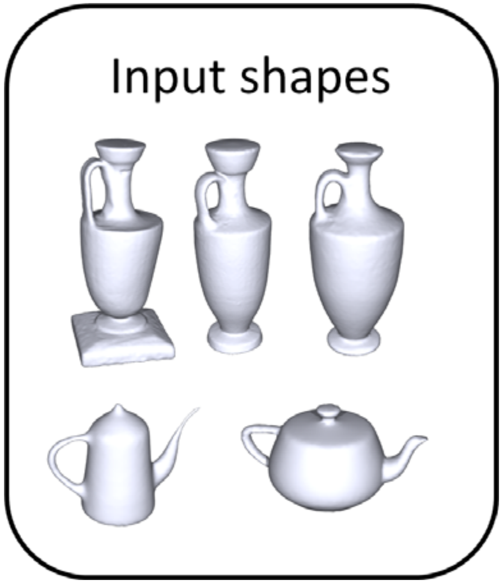
Similar



Less similar

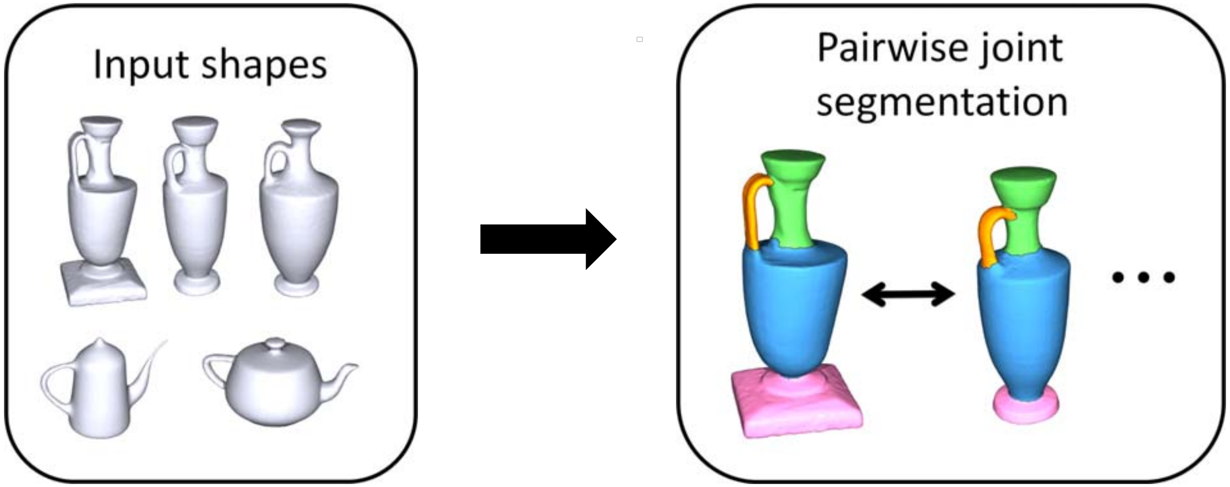
Multi-way joint segmentation

- Input shapes
  - Different objects
  - Different categories



Multi-way joint segmentation

- Perform all pair-wise joint segmentation to determine pairs of similar shapes



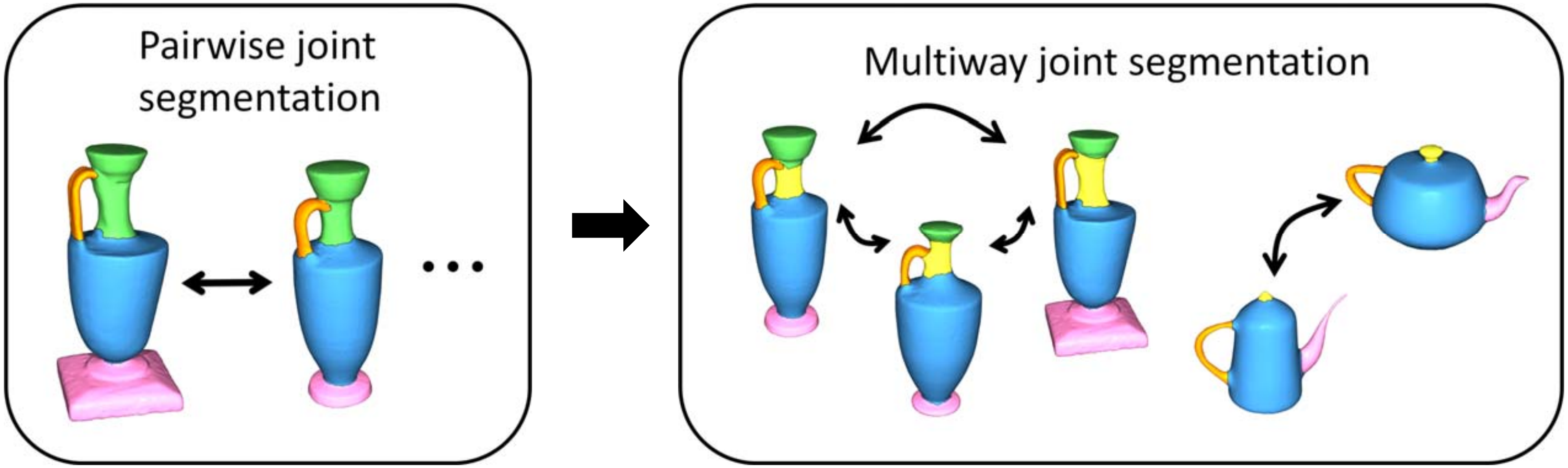
Input shapes

Pairwise joint segmentation

Multi-way joint segmentation

- Objective function

$$\sum_{i=1}^{n} \text{score}(S_i) + \sum_{(S_i,S_j)\in\mathcal{E}} \text{consistency}(S_i, S_j)$$

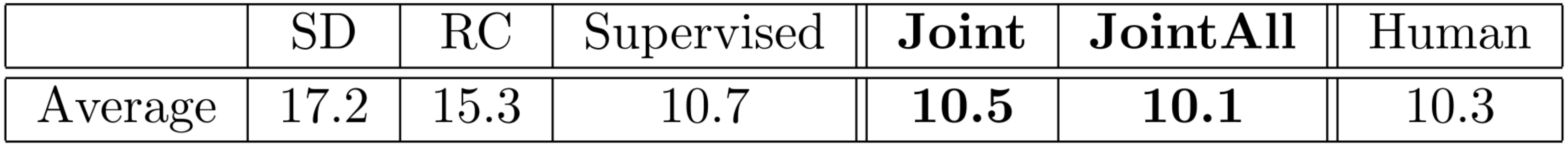


Princeton Segmentation Benchmark [Chen et al. 09]

Joint : Joint shape segmentation per each category

JointAll : Joint shape segmentation over the entire database

Rand index metric [Rand 1971] - the smaller, the better



	SD	RC	Supervised	**Joint**	**JointAll**	Human
Average	17.2	15.3	10.7	**10.5**	**10.1**	10.3

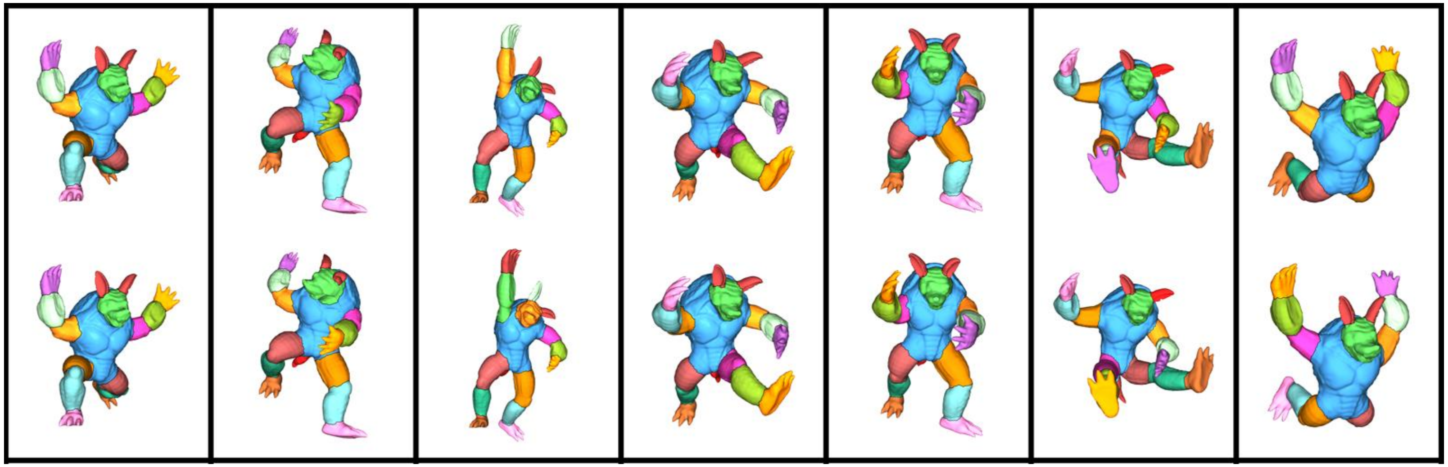
- Significantly better than single shape segmentations
- Competitive against supervised segmentation
- JointAll is slightly better than Joint

Rand Index Scores on PSB [Chen et.al 09]

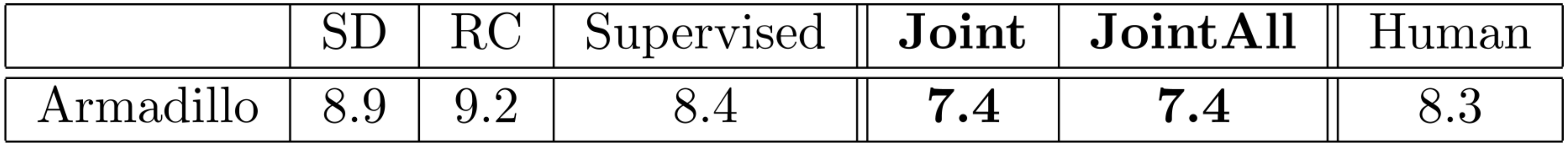
When shape variation of the input is big

Top: Joint

Bottom: JointAll



• • •



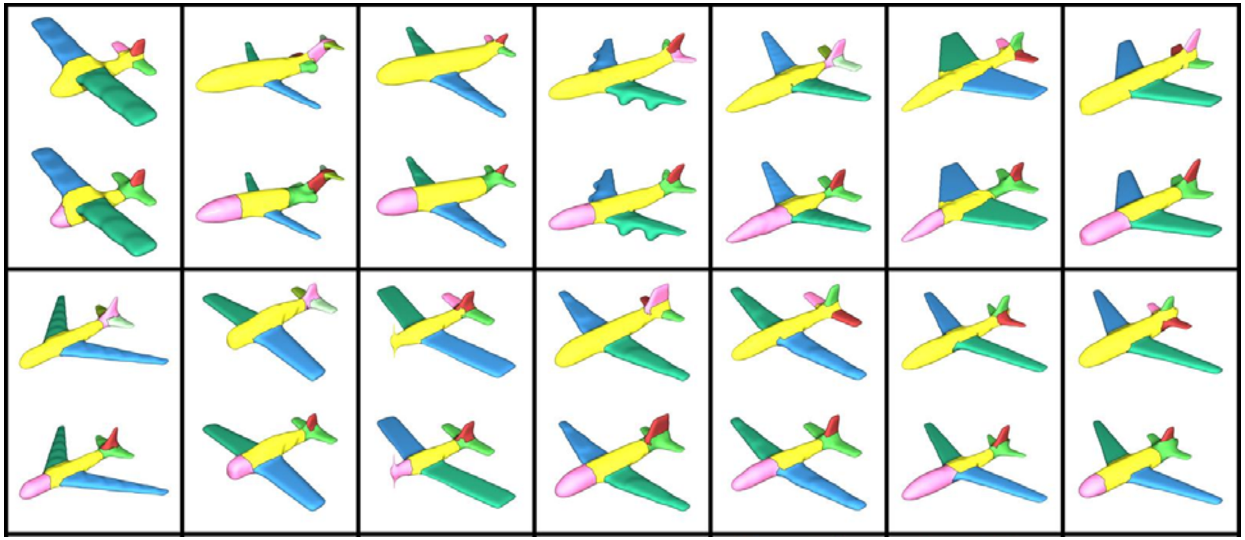
	SD	RC	Supervised	**Joint**	**JointAll**	Human
Armadillo	8.9	9.2	8.4	**7.4**	**7.4**	8.3

Rand Index Scores on PSB [Chen et.al 09]

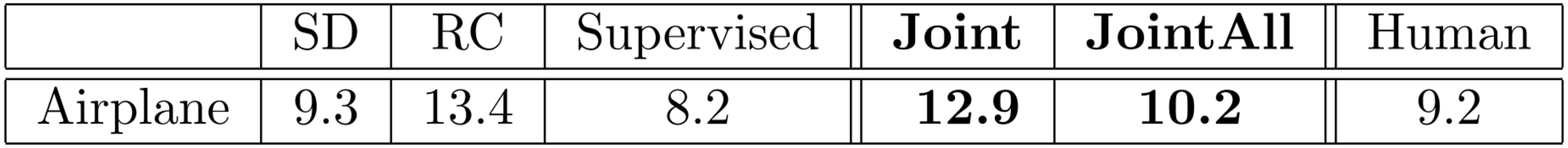
When shape variation of the input is small

Top: Joint

Bottom: JointAll



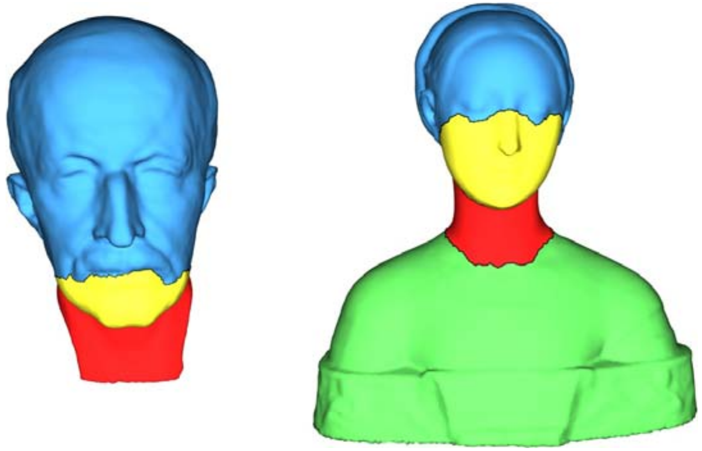
...



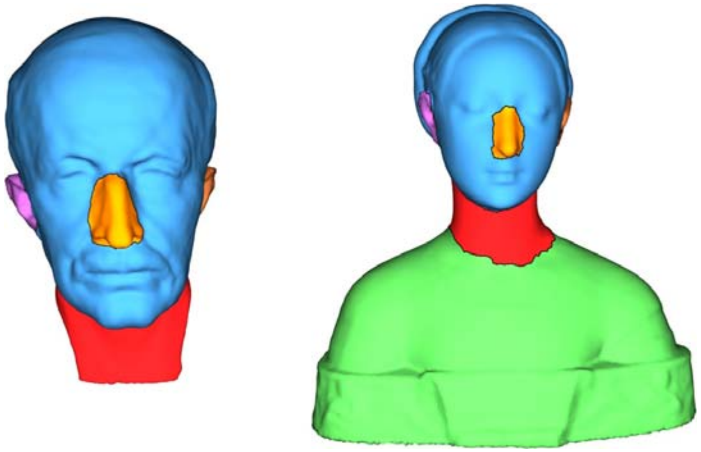
	SD	RC	Supervised	**Joint**	**JointAll**	Human
Airplane	9.3	13.4	8.2	**12.9**	**10.2**	9.2

Versus Supervised Method [Kalogerakis et al.10]

Supervised segmentation



Joint shape segmentation



Summary

- Single-shape segmentations are limited
  - No algorithm is suitable for any shape categories
- Data-driven shape segmentations can improve segmentation quality
- The behavior of supervised method and unsupervised method is different
  - Supervised method requires shapes to be similar to each other
  - Unsupervised method requires variation in shapes

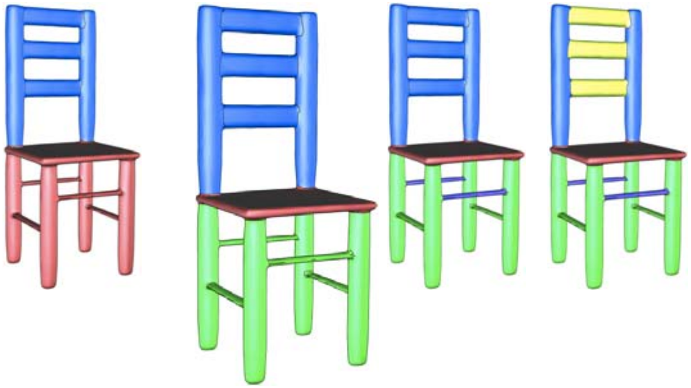
Future directions

- Hierarchical segmentation
- Man-made objects

Single-Level Versus Hierarchical

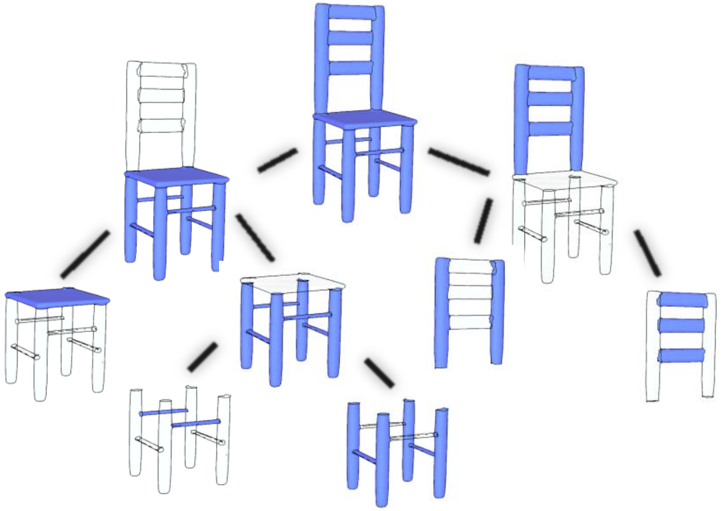
Single level

[Chen et al. 09, Kalogerkis et al. 11,
Huang et al.11, Sidi et al.11,...]



Hierarchical

[Martinet 2007, Wang et al. 11]



- Hierarchical representations
  - Less ambiguous than single level representation
  - Discrete scale-space representation

  …

Architectural Models