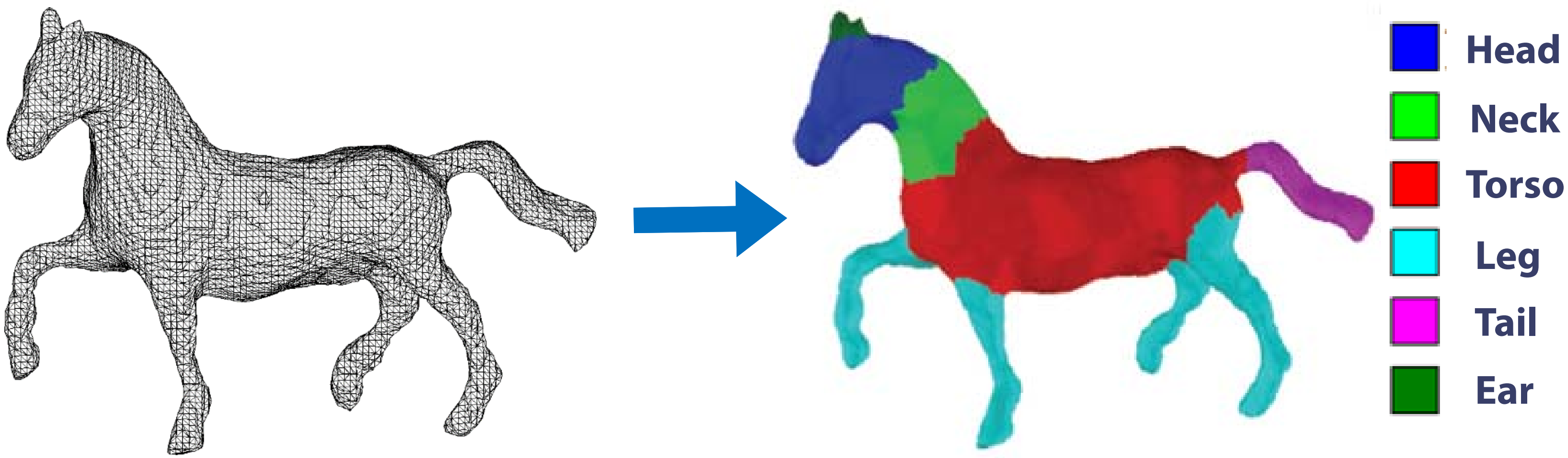
Data-driven 3D shape analysis and synthesis



Evangelos Kalogerakis
UMass Amherst

3D shapes for computer-aided design

Architecture

Interior design

3D shapes for information visualization

Geo-visualization

Scientific visualization

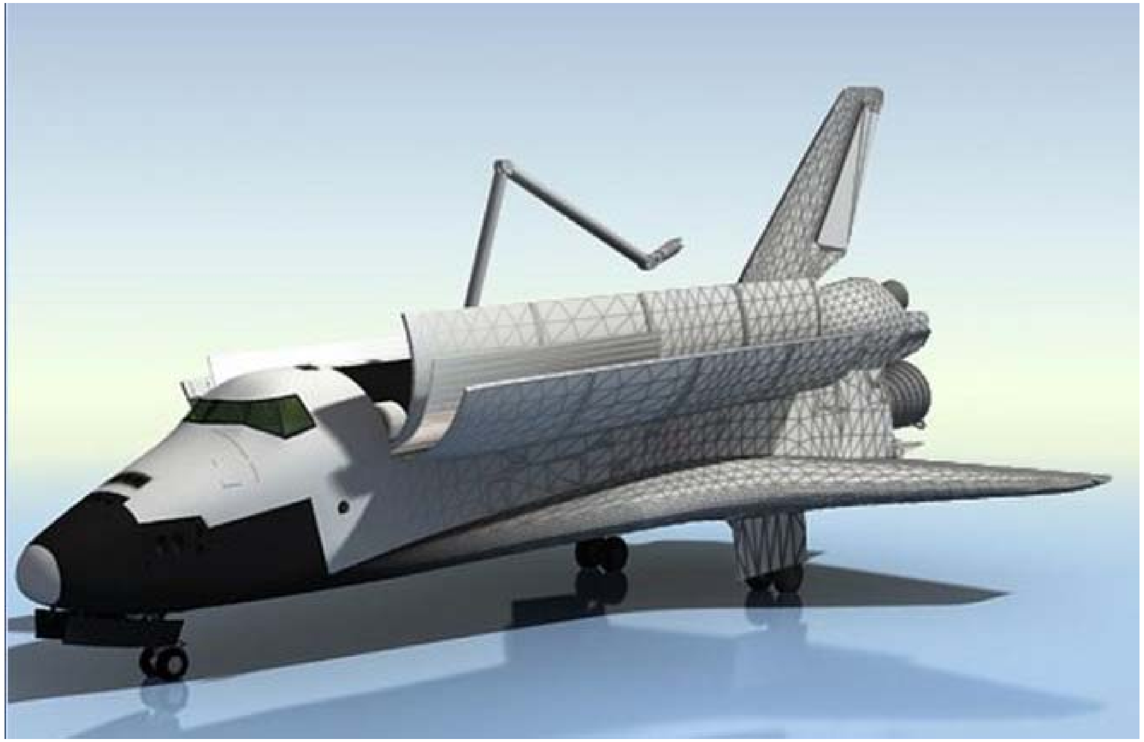
3D shapes for digital entertainment

Video games

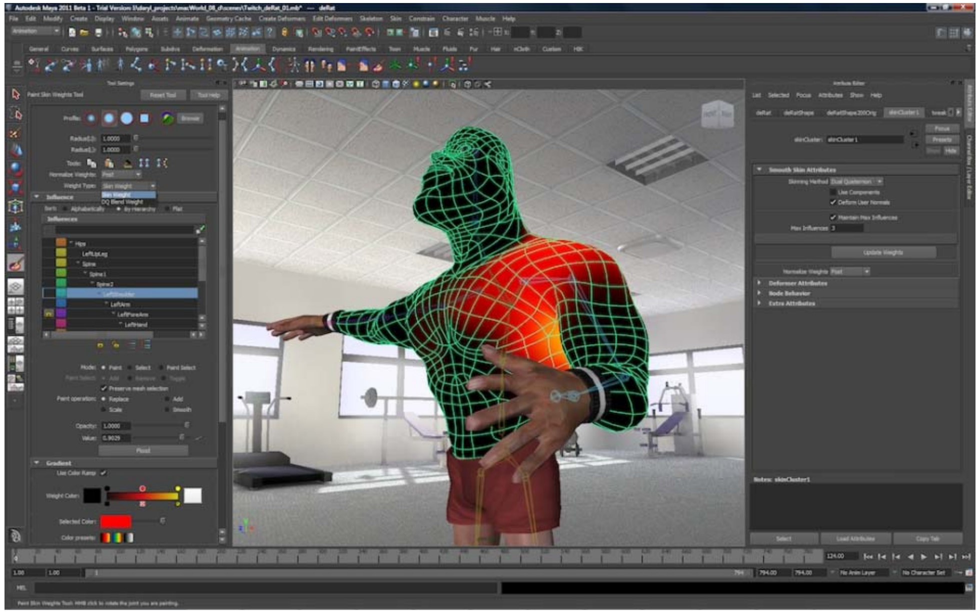
Digital representations of 3D shapes

Polygon mesh

Digital representations of 3D shapes

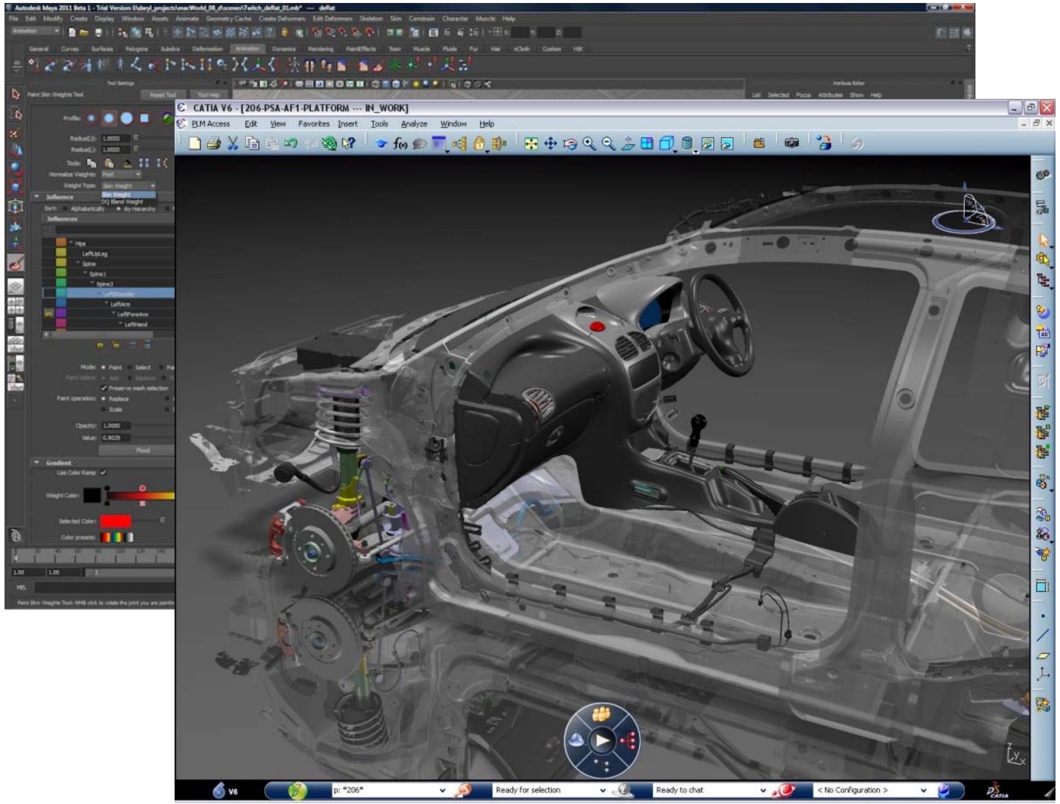


Digitizing our imagination



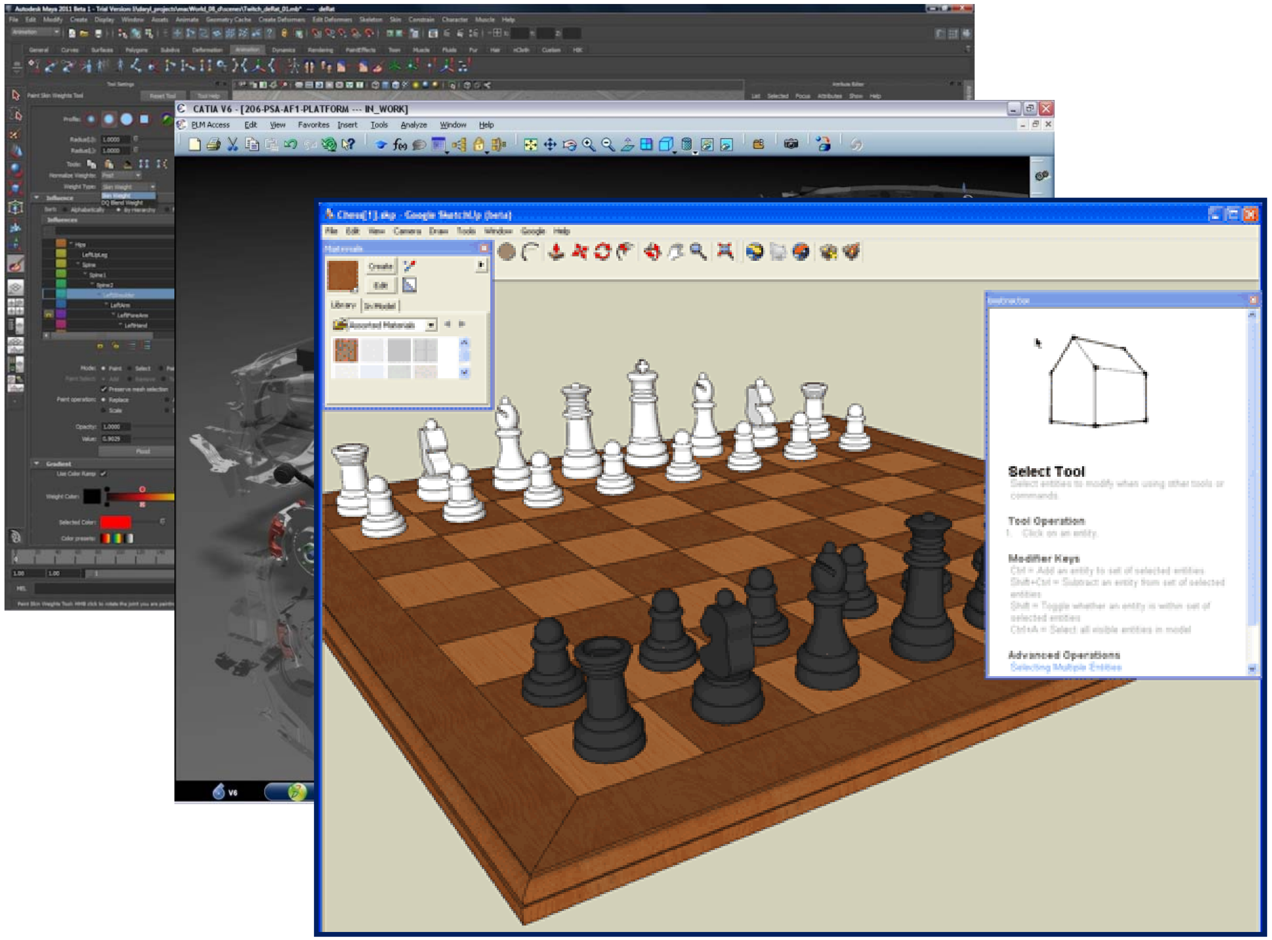
Professional 3D modeling tools [Autodesk Maya]

Digitizing our imagination



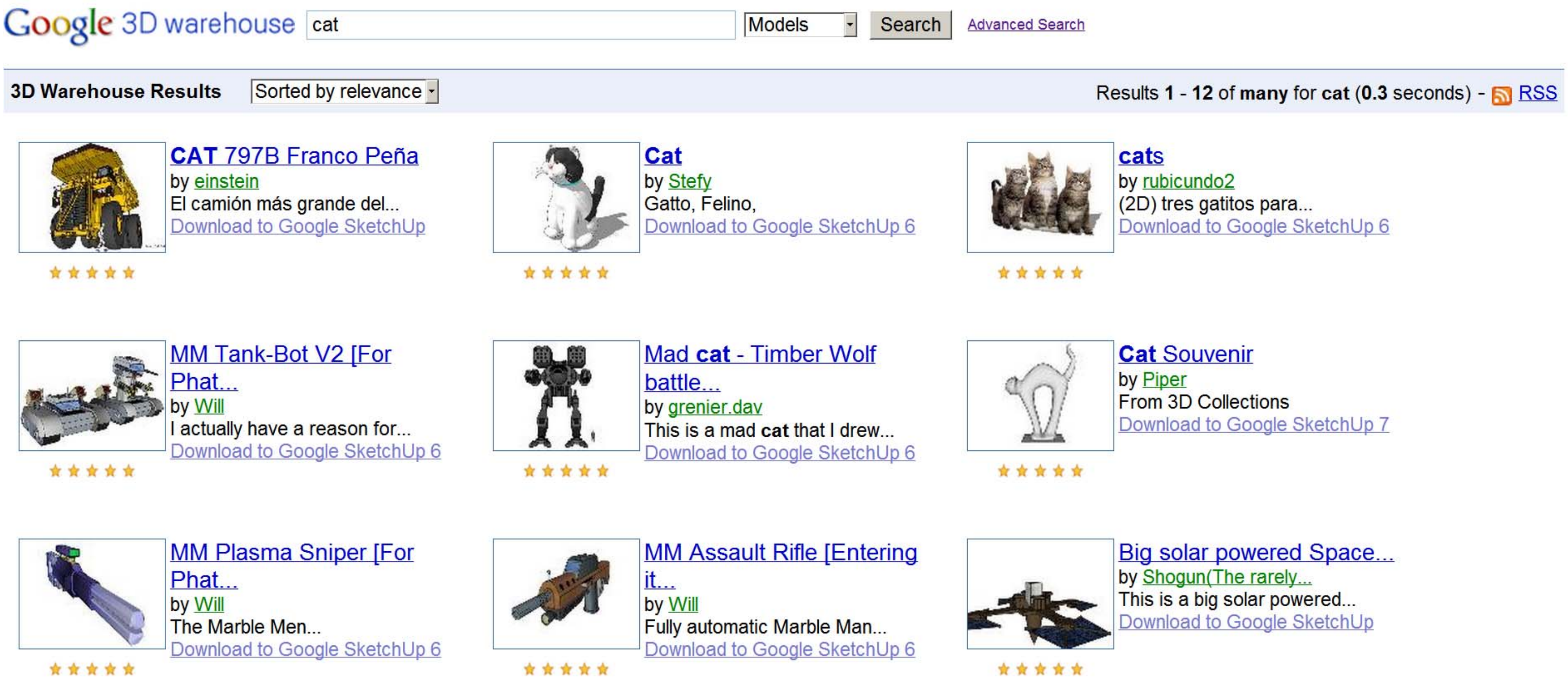
Computer-Aided Design tools [Catia]

Digitizing our imagination



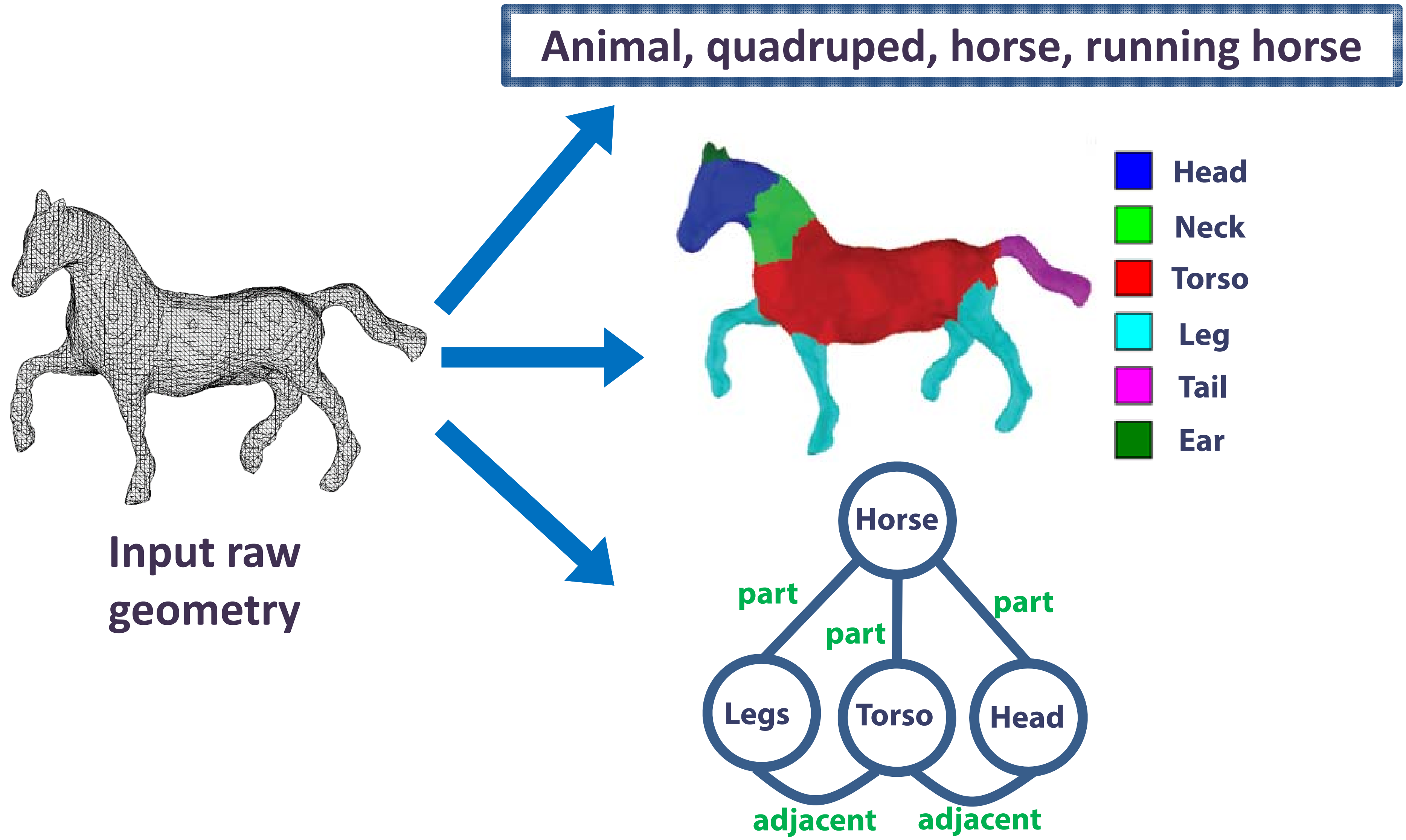
General-Purpose Modeling tools
[Google Sketch-up]

3D shape repositories

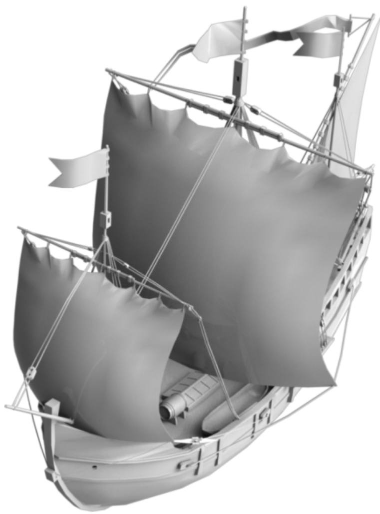


[Google 3D Warehouse]

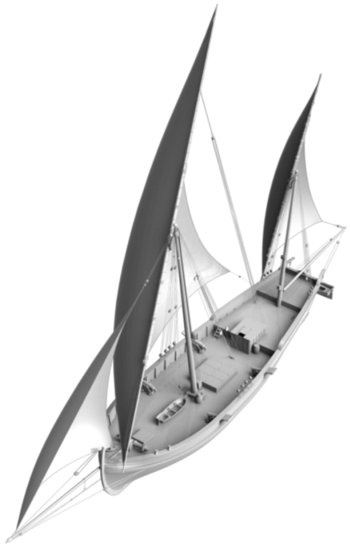
Shape understanding



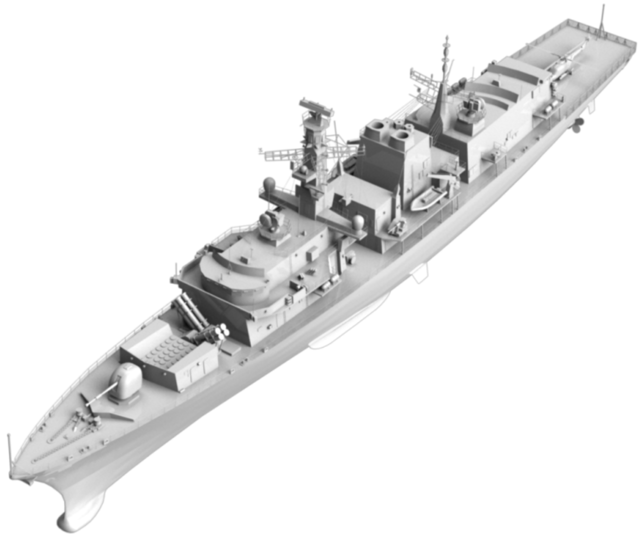
Why shape understanding?
Shape categorization



Sailing Ship,
Galleon



Sailing ship,
Yawl

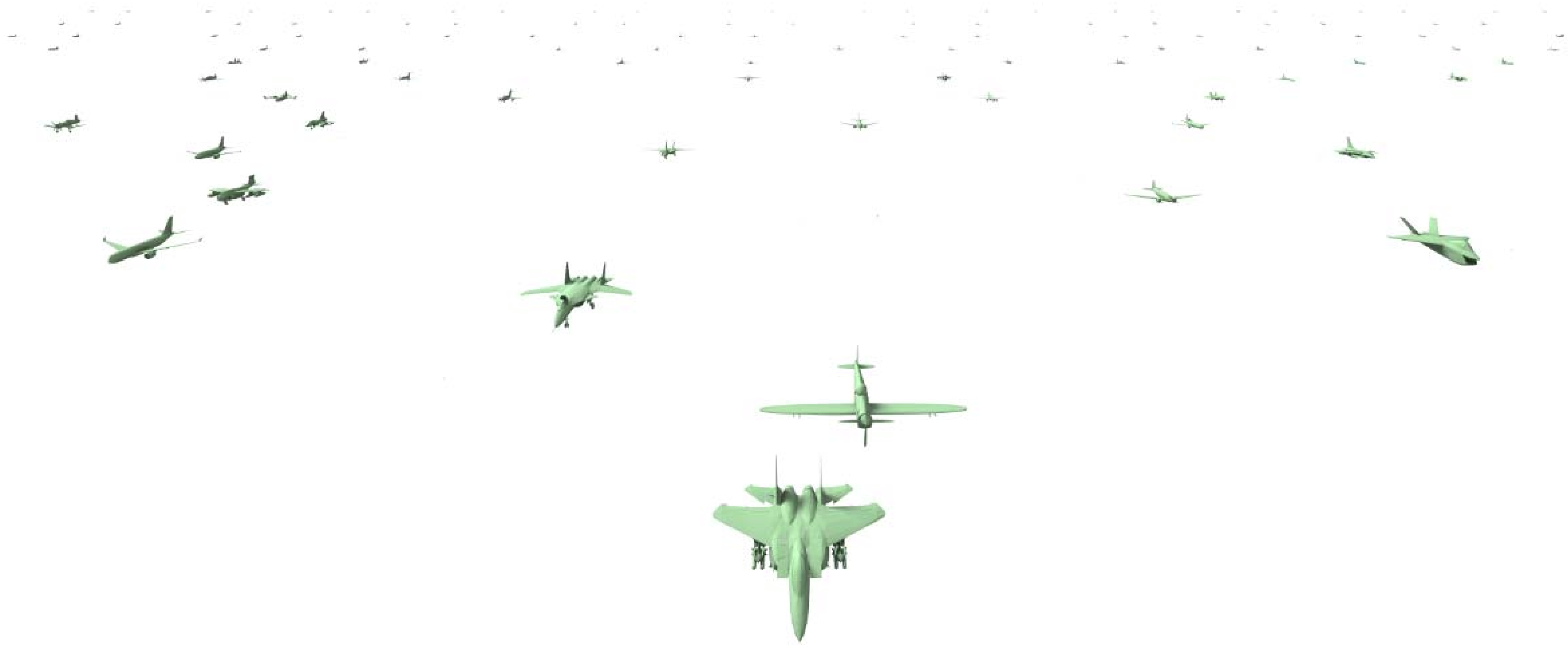


Military ship,
Frigate

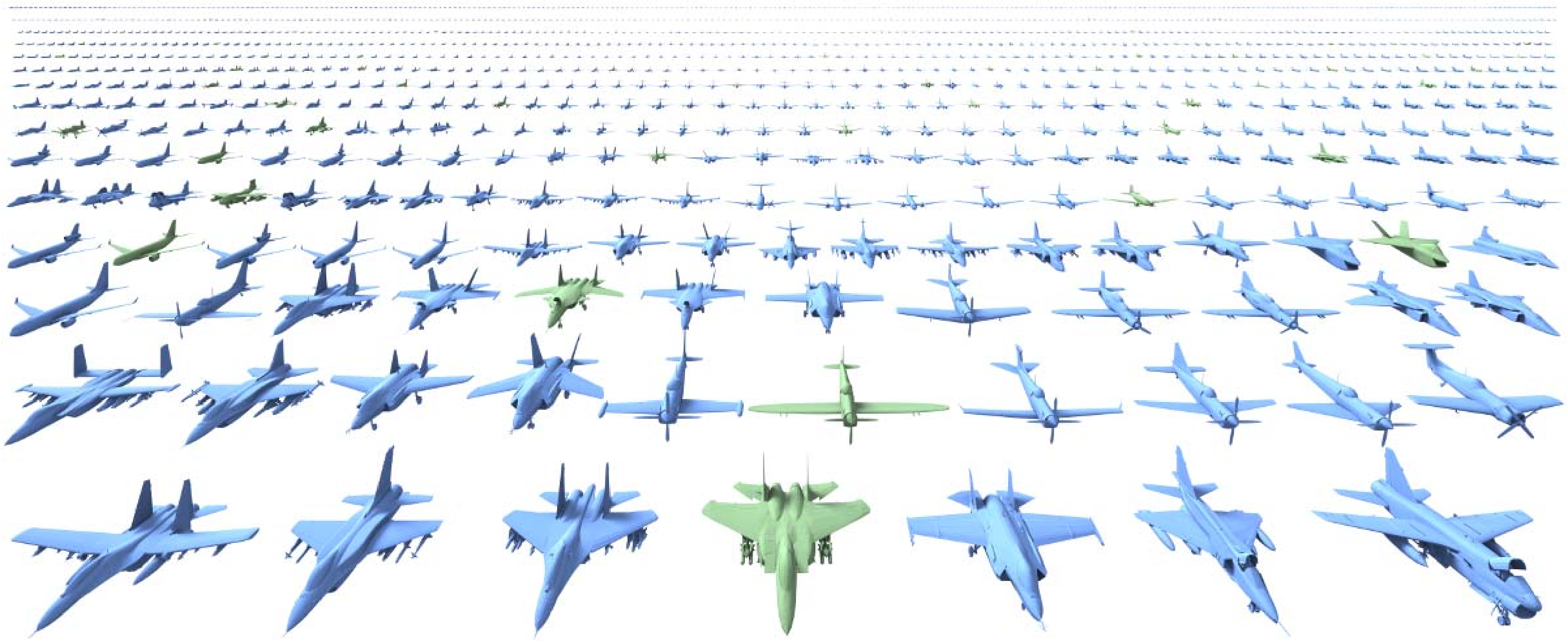
Why shape understanding?
3D Modeling

Chaudhuri, Kalogerakis, Guibas, Koltun, SIGGRAPH 2011

Why shape understanding?
Shape synthesis

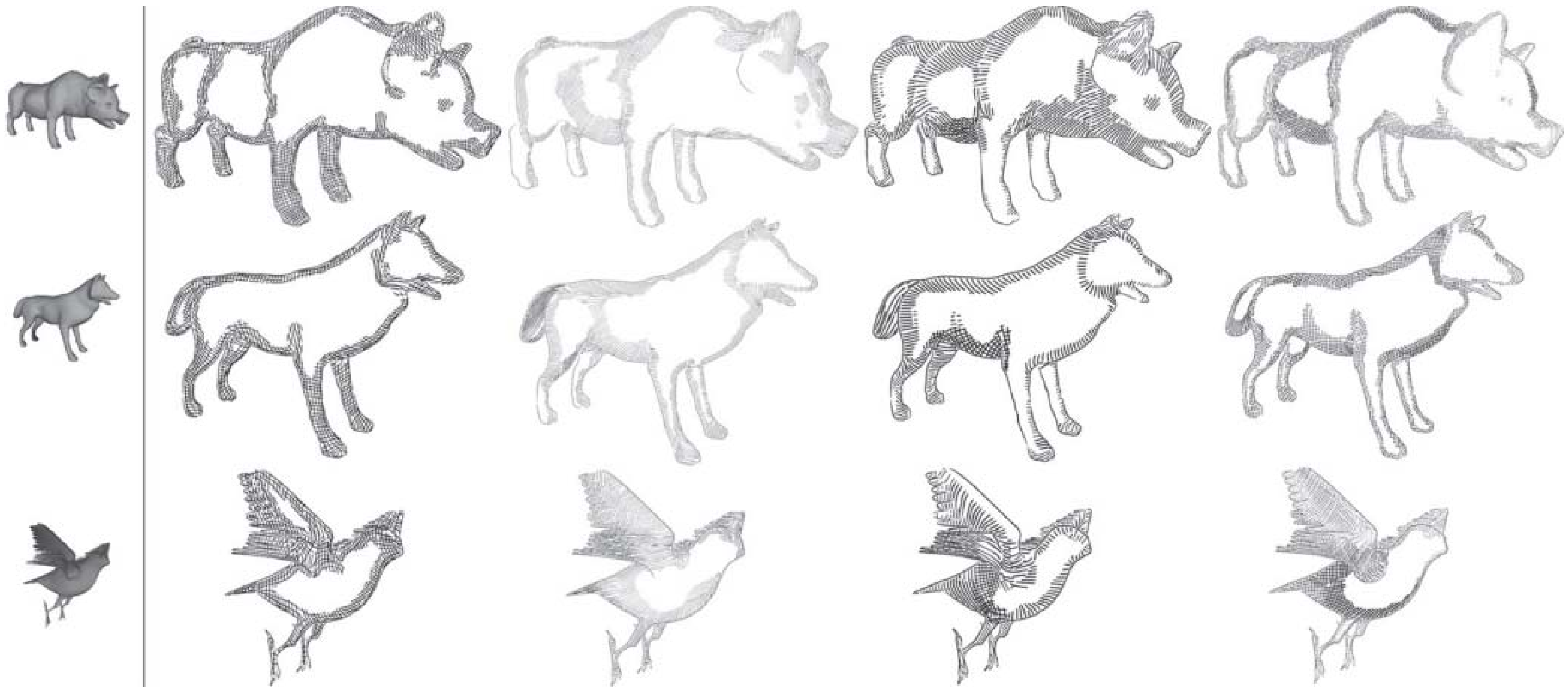


Why shape understanding?
Shape synthesis



Kalogerakis, Chaudhuri, Koller, Koltun, SIGGRAPH 2012

Why shape understanding?
Artistic rendering

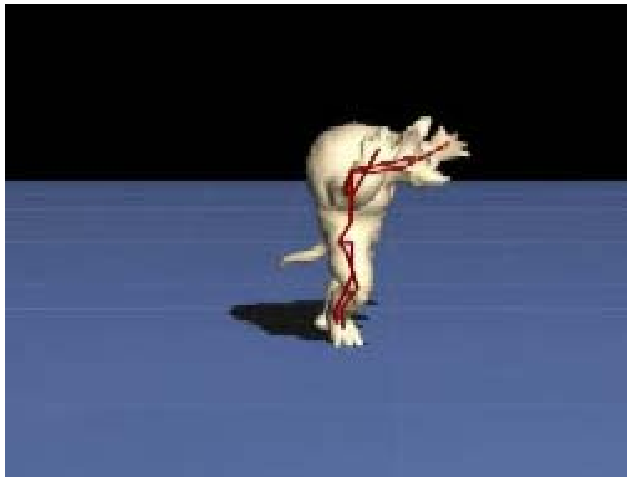


Kalogerakis, Nowrouzehahrai, Breslav, Hertzmann, TOG 2012

Why shape understanding?
Texturing

Kalogerakis, Hertzmann, Singh, SIGGRAPH 2010

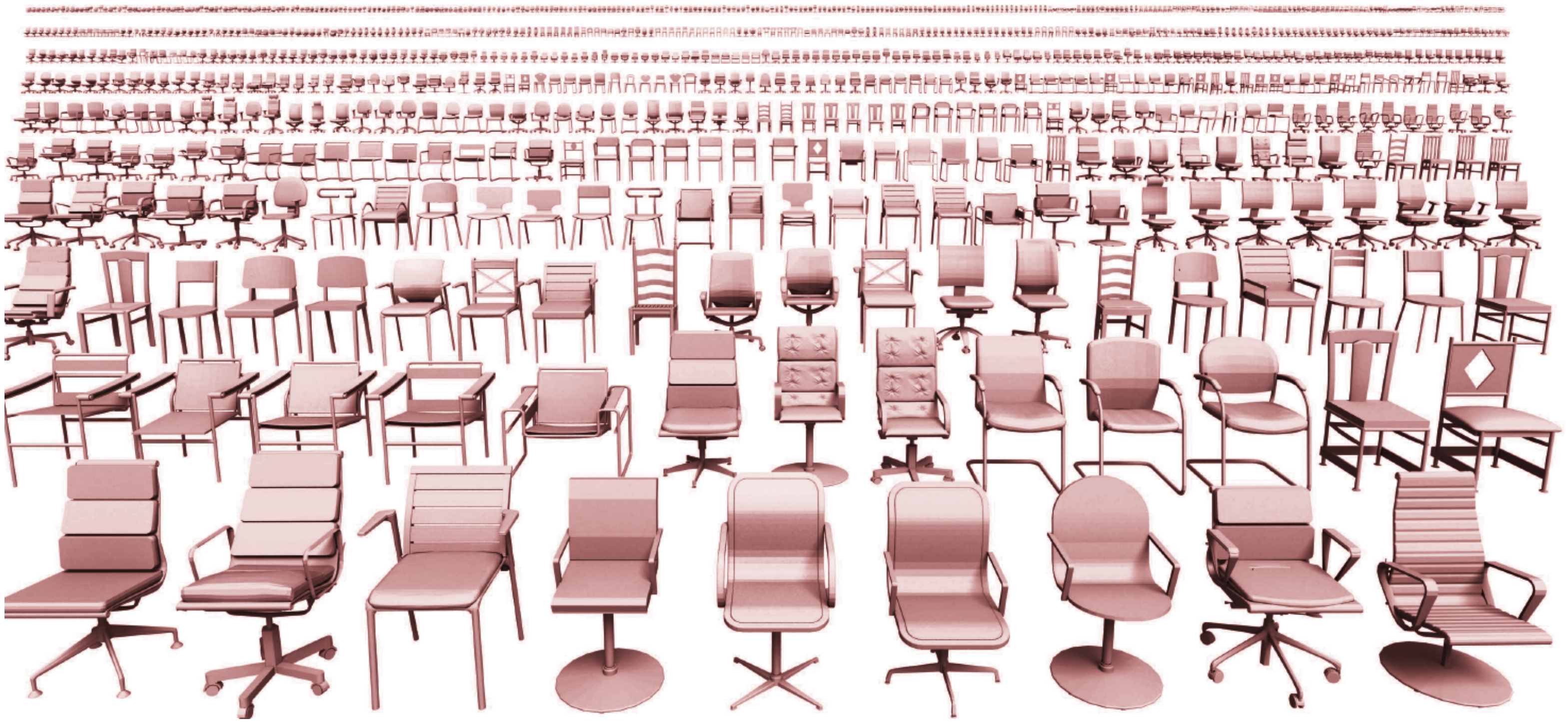
Why shape understanding?
Character Animation



Simari, Nowrouzezahrai, Kalogerakis, Singh, SGP 2009

How can we perform shape understanding?

It is extremely hard to perform shape understanding with a set of deterministic, manually specified rules!



Key idea: probabilistic models for shapes

Define a probability distribution over high-level shape attributes given geometry (**discriminative approach**), or both (**generative approach**).

Learn this distribution by combining training data and expert knowledge.

Efficiently infer unknown attributes given observed evidence.

First part of my talk:
Learning 3D shape segmentation and labeling

Learning 3D Mesh Segmentation and Labeling

Evangelos Kalogerakis Aaron Hertzmann Karan Singh
University of Toronto

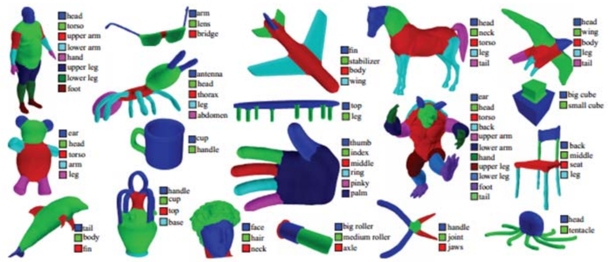


Figure 1: *Labeling and segmentation results from applying our algorithm to one mesh each from every category in the Princeton Segmentation Benchmark [Chen et al. 2009]. For each result, the algorithm was trained on the other meshes in the same class, e.g., the human was labeled after training on the other meshes in the human class.*

Abstract

This paper presents a data-driven approach to simultaneous segmentation and labeling of parts in 3D meshes. An objective function is formulated as a Conditional Random Field model, with terms assessing the consistency of faces with labels, and terms between labels of neighboring faces. The objective function is learned from a collection of labeled training meshes. The algorithm uses hundreds of geometric and contextual label features and learns different types of segmentations for different tasks, without requiring manual parameter tuning. Our algorithm achieves a significant improvement in results over the state-of-the-art when evaluated on the Princeton Segmentation Benchmark, often producing segmentations and labelings comparable to those produced by humans.

1 Introduction

Segmentation and labeling of 3D shapes into meaningful parts is fundamental to shape understanding and processing. Numerous tasks in geometric modeling, manufacturing, animation and texturing of 3D meshes rely on their segmentation into parts. Many of these problems further require labeled segmentations, where the parts are also recognized as instances of known part types. For most of these applications, the segmentation and labeling of the input shape is manually specified. For example, to synthesize texture for a humanoid mesh, one must identify which parts should have "arm" texture, which should have "leg" texture, and so on. Even tasks such as 3D shape matching or retrieval, which do not directly require labeled-segmentations, could benefit from knowledge of constituent parts and labels. However, there has been very little research in part labeling for 3D meshes, and 3D object segmentation likewise remains an open research problem [Chen et al. 2009].

This paper introduces a data-driven approach to simultaneous segmentation and labeling of parts in 3D meshes. Labeling of mesh parts is expressed as a problem of optimizing a Conditional Random Field (CRF) [Lafferty et al. 2001]. This segments a mesh into parts, with each part having a corresponding label. The CRF objective function includes unary terms that assess the consistency of faces with labels, and pairwise terms between labels of adjacent faces. The objective function is learned from a collection of labeled training meshes. The basic terms of the CRF are learned using JointBoost classifiers [Torralba et al. 2007], which automatically select from among hundreds of possible geometric features to choose those that are relevant for a particular segmentation task. Holdout validation is used to learn additional CRF parameters. We evaluate our methods on the Princeton Segmentation Benchmark, with manually-added labels. Our method yields 94% labeling accuracy, and is the first labeling method applicable to such a broad range of meshes. In segmentation, our method yields 9.5% Rand In-

http://www.dgp.toronto.edu/~kalo/papers/LabelMeshes/

<u>**Contributions:**</u>

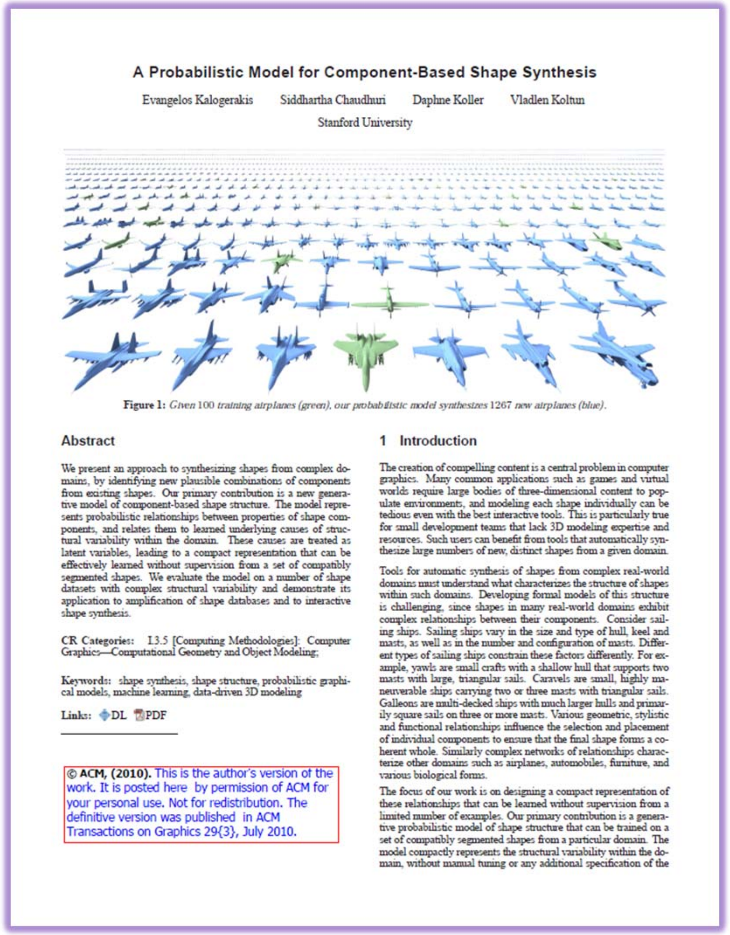
Segmentation and labeling of parts with a prob. discriminative model

Major improvements over prior work

Data-driven, learnt from examples

Kalogerakis, Hertzmann, Singh, SIGGRAPH 2010

Second part of my talk:
A generative model of shapes



A Probabilistic Model for Component-Based Shape Synthesis

Evangelos Kalogerakis Siddhartha Chaudhuri Daphne Koller Vladlen Koltun
Stanford University

Figure 1: *Given 100 training airplanes (green), our probabilistic model synthesizes 1267 new airplanes (blue).*

Abstract

We present an approach to synthesizing shapes from complex domains, by identifying new plausible combinations of components from existing shapes. Our primary contribution is a new generative model of component-based shape structure. The model represents probabilistic relationships between properties of shape components, and relates them to learned underlying causes of structural variability within the domain. These causes are treated as latent variables, leading to a compact representation that can be effectively learned without supervision from a set of compatibly segmented shapes. We evaluate the model on a number of shape datasets with complex structural variability and demonstrate its application to amplification of shape databases and to interactive shape synthesis.

CR Categories: I.3.5 [Computing Methodologies]: Computer Graphics—Computational Geometry and Object Modeling;

Keywords: shape synthesis, shape structure, probabilistic graphical models, machine learning, data-driven 3D modeling

Links: DL PDF

© ACM, (2010). This is the author's version of the work. It is posted here by permission of ACM for your personal use. Not for redistribution. The definitive version was published in ACM Transactions on Graphics 29{3}, July 2010.

1 Introduction

The creation of compelling content is a central problem in computer graphics. Many common applications such as games and virtual worlds require large bodies of three-dimensional content to populate environments, and modeling each shape individually can be tedious even with the best interactive tools. This is particularly true for small development teams that lack 3D modeling expertise and resources. Such users can benefit from tools that automatically synthesize large numbers of new, distinct shapes from a given domain.

Tools for automatic synthesis of shapes from complex real-world domains must understand what characterizes the structure of shapes within such domains. Developing formal models of this structure is challenging, since shapes in many real-world domains exhibit complex relationships between their components. Consider sailing ships. Sailing ships vary in the size and type of hull, keel and masts, as well as in the number and configuration of masts. Different types of sailing ships constrain these factors differently. For example, yawls are small crafts with a shallow hull that supports two masts with large, triangular sails. Caravels are small, highly maneuverable ships carrying two or three masts with triangular sails. Galleons are multi-decked ships with much larger hulls and primarily square sails on three or more masts. Various geometric, stylistic and functional relationships influence the selection and placement of individual components to ensure that the final shape forms a coherent whole. Similarly complex networks of relationships characterize other domains such as airplanes, automobiles, furniture, and various biological forms.

The focus of our work is on designing a compact representation of these relationships that can be learned without supervision from a limited number of examples. Our primary contribution is a generative probabilistic model of shape structure that can be trained on a set of compatibly segmented shapes from a particular domain. The model compactly represents the structural variability within the domain, without manual tuning or any additional specification of the

<u>Contributions:</u>

Learns structural variability in 3D shapes

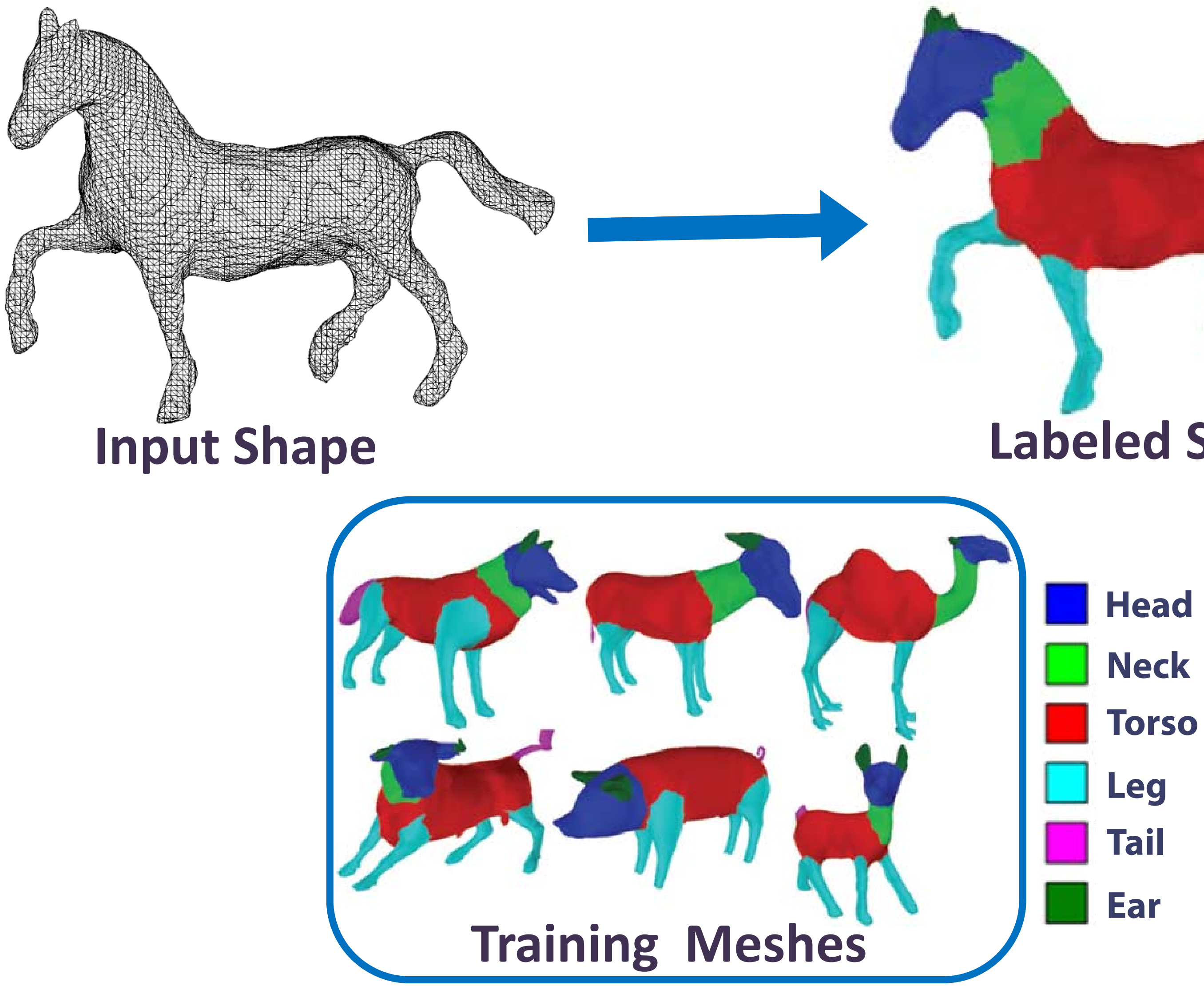
Automatic shape synthesis in complex domains (airplanes, ships, furniture, game characters)

Kalogerakis, Chaudhuri, Koller, Koltun, SIGGRAPH 2012

Outline

1. **Learning 3D shape segmentation and labeling**
   [Kalogerakis et al., SIGGRAPH 2010]
2. **A generative model of shapes**
3. **Other ML applications to graphics and vision**
4. **Future work**

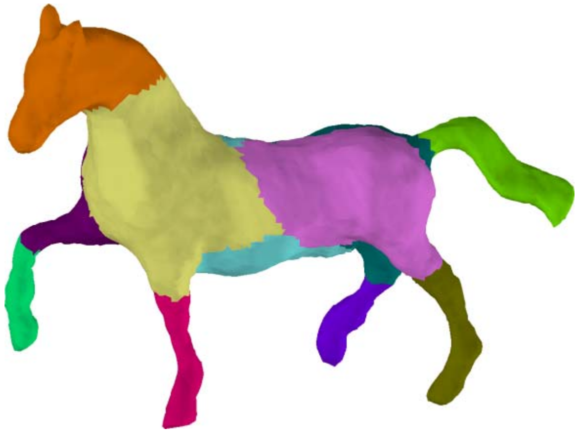
Goal: shape segmentation and labeling



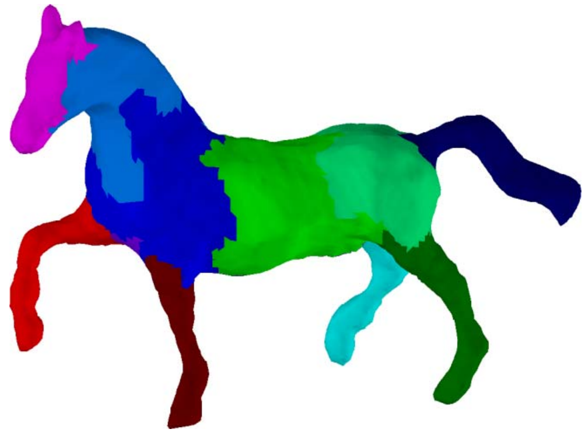
Related work: mesh segmentation

Shape Diameter
[Shapira et al. 2010]

Randomized Cuts
[Golovinskiy and Funkhouser 2008]

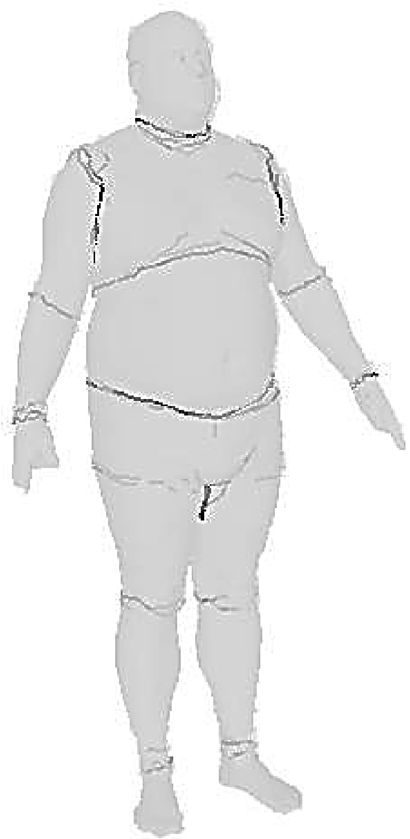


Random Walks
[Lai et al. 2008]



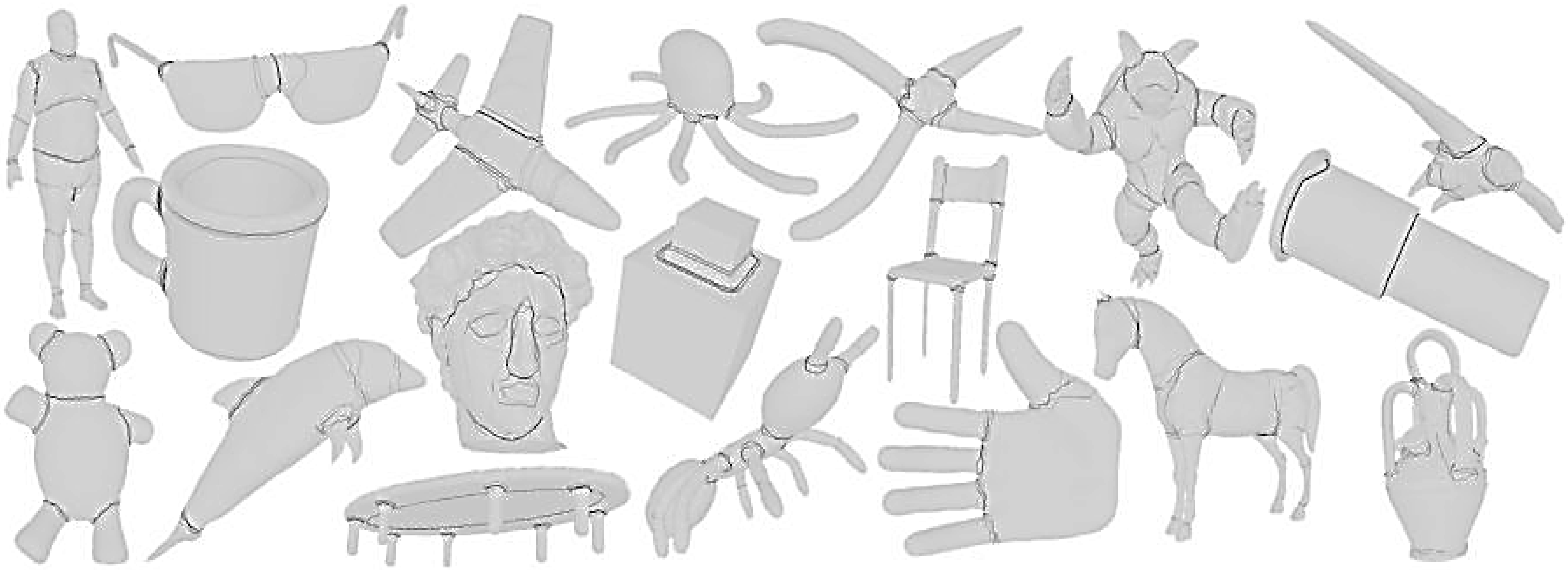
Fitting Primitives
[Attene et al. 2006]

Is human-level shape analysis possible without using prior knowledge?



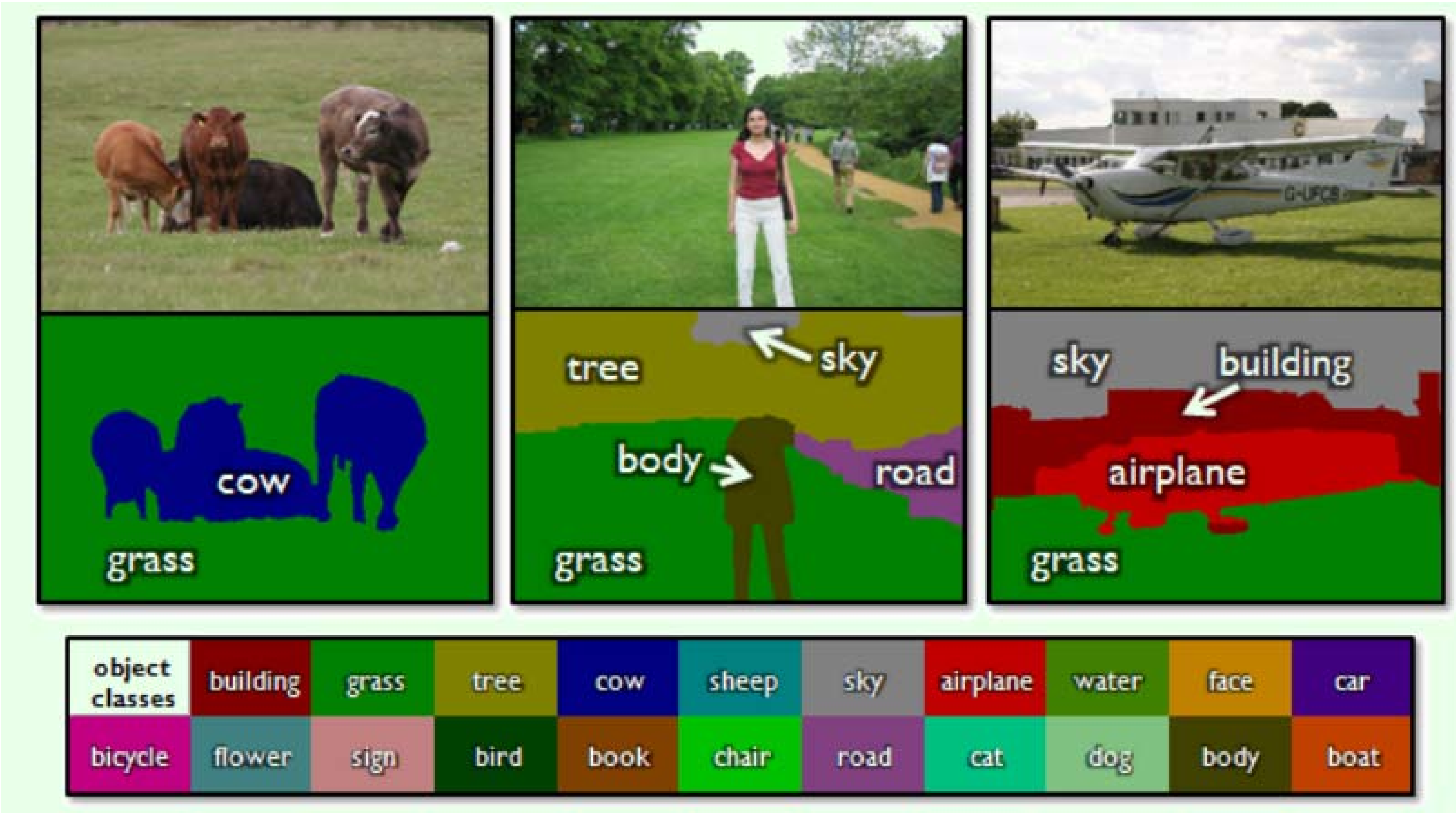
[X. Chen et al. SIGGRAPH 2009]

Must we hand-tune algorithms for each type of shape?



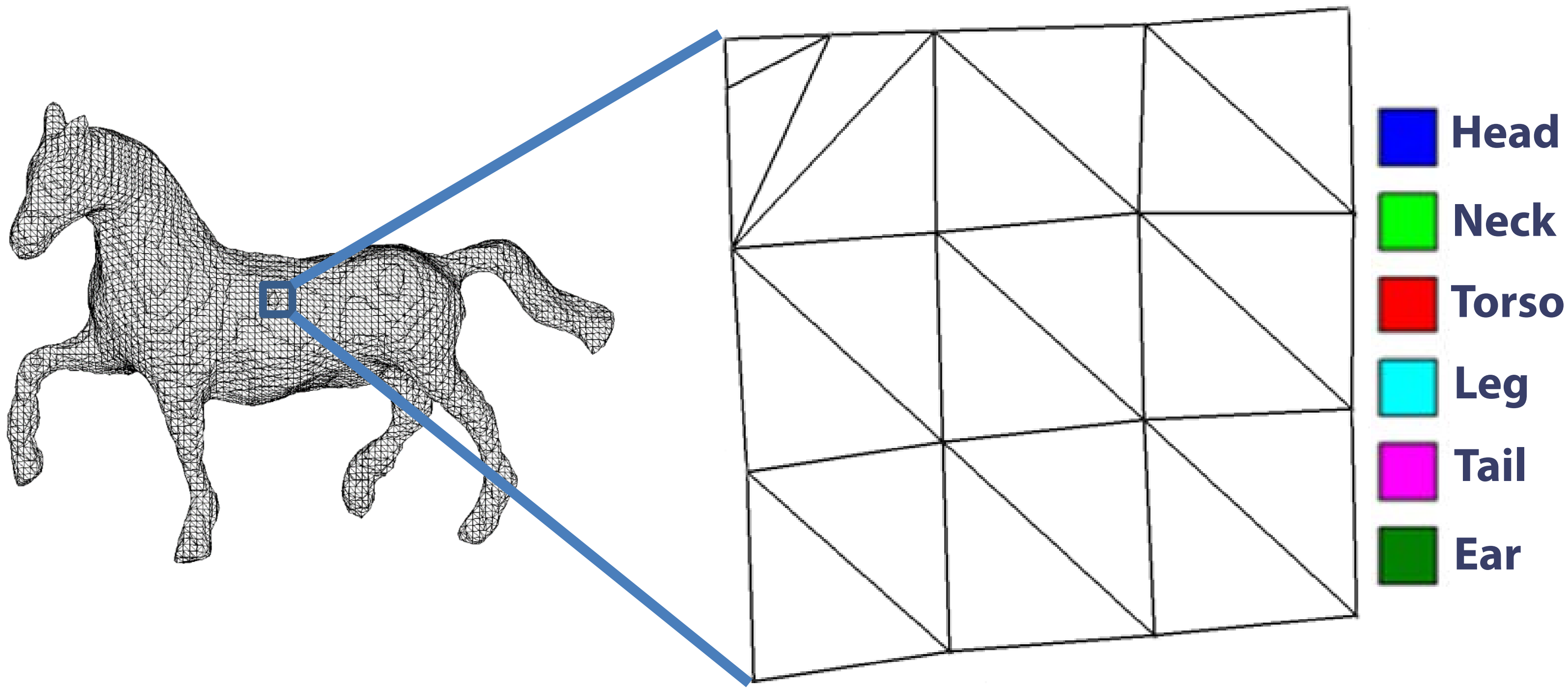
[X. Chen et al. SIGGRAPH 2009]

Related work: image segmentation and labeling

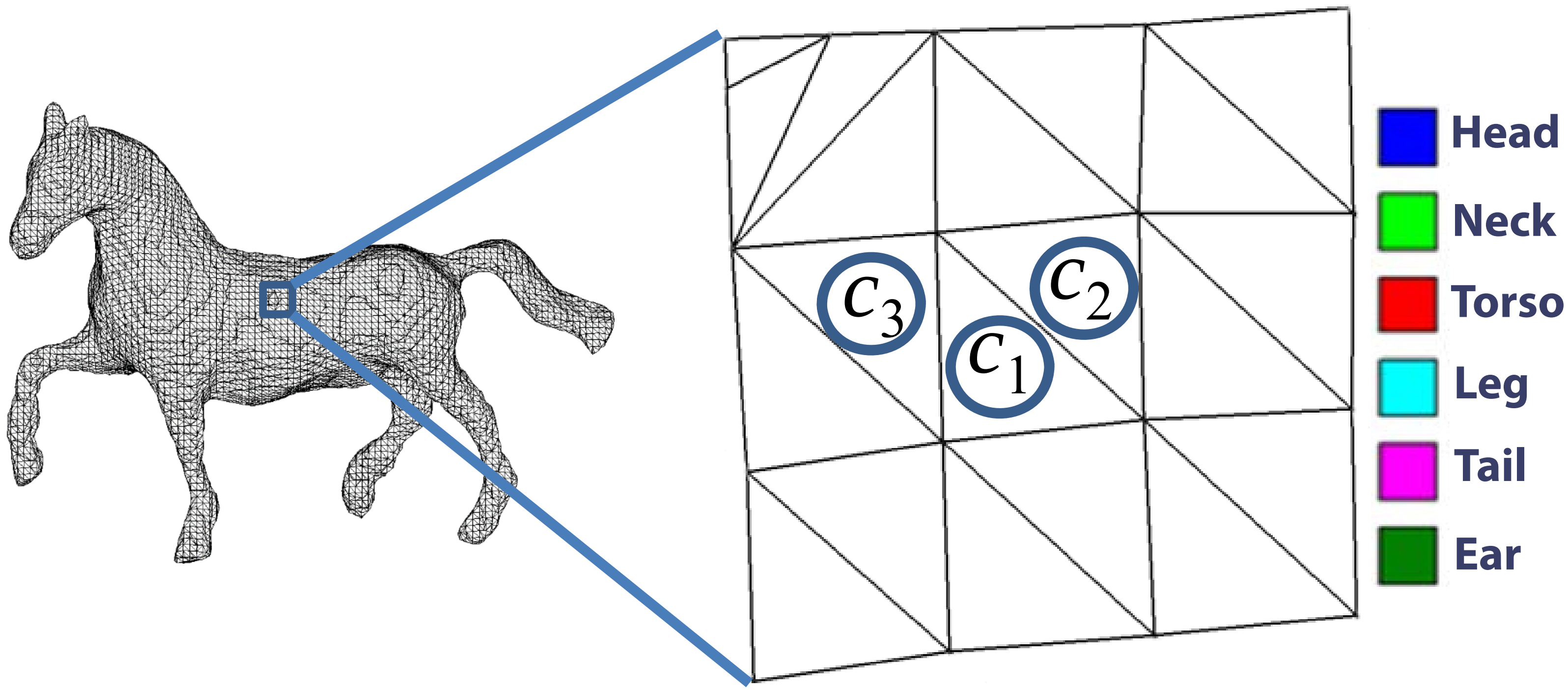


Textonboost
[Shotton et al. ECCV 2006]

Labeling problem statement



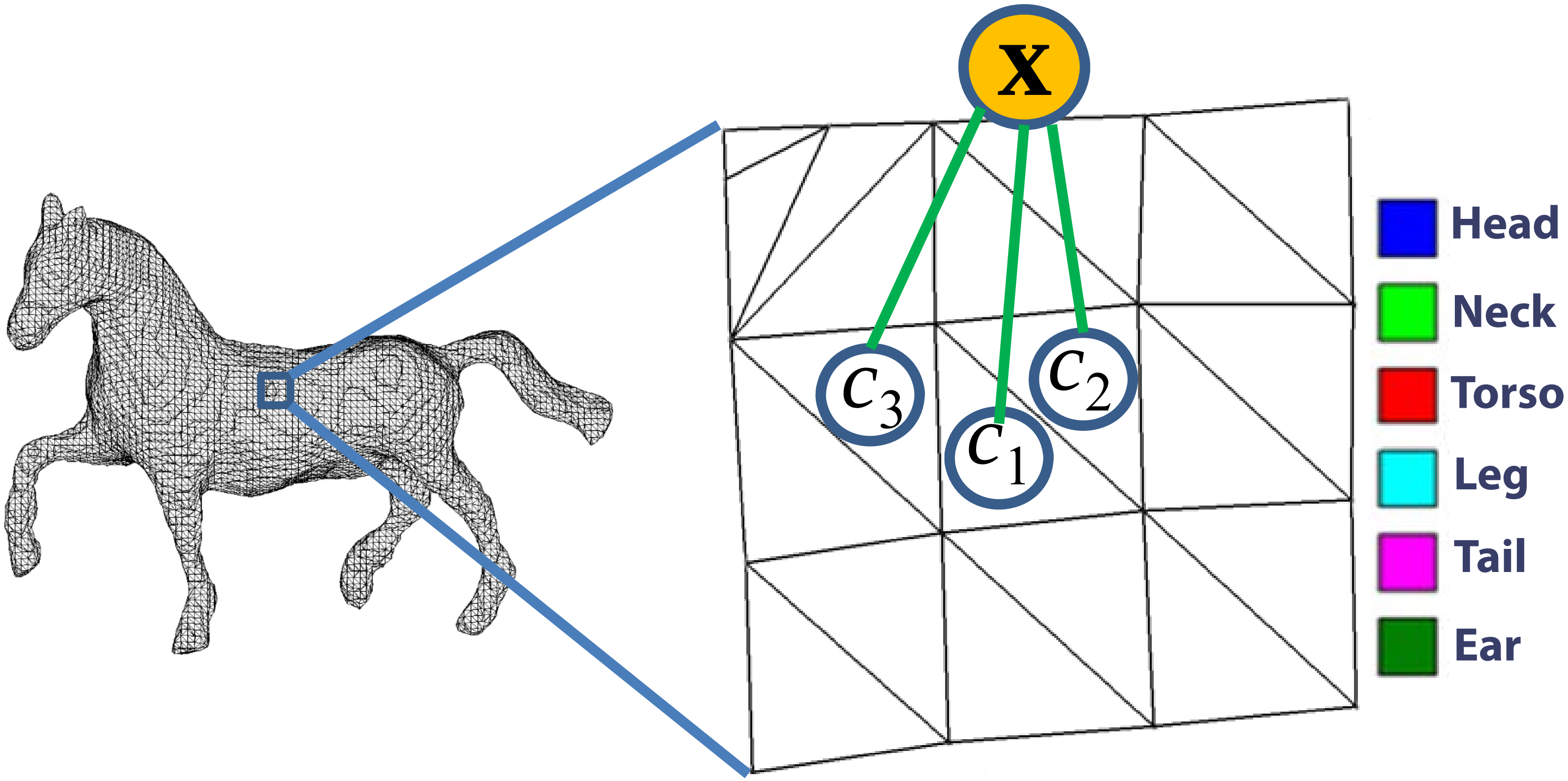
Labeling problem statement



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

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

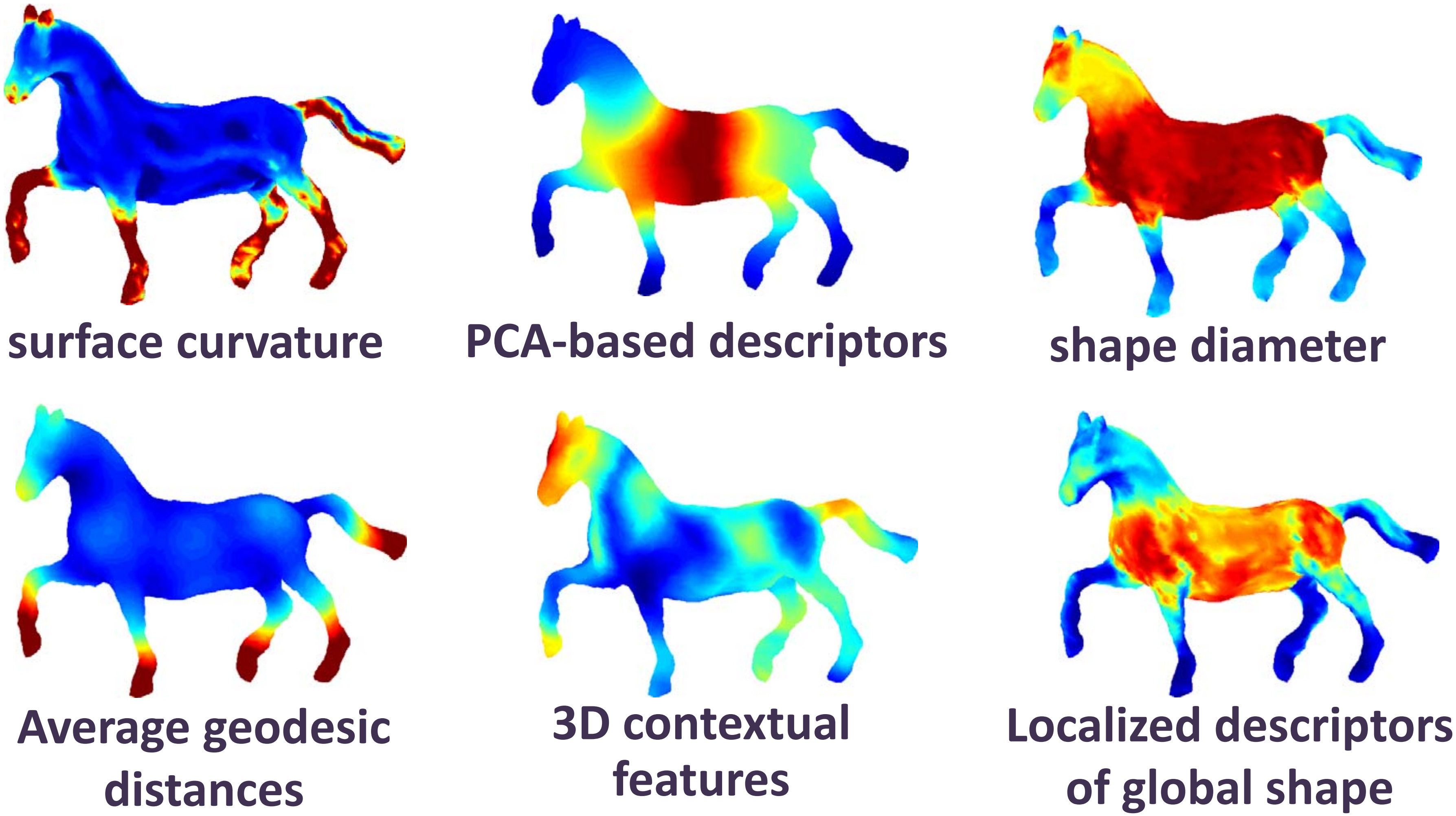
Labeling problem statement



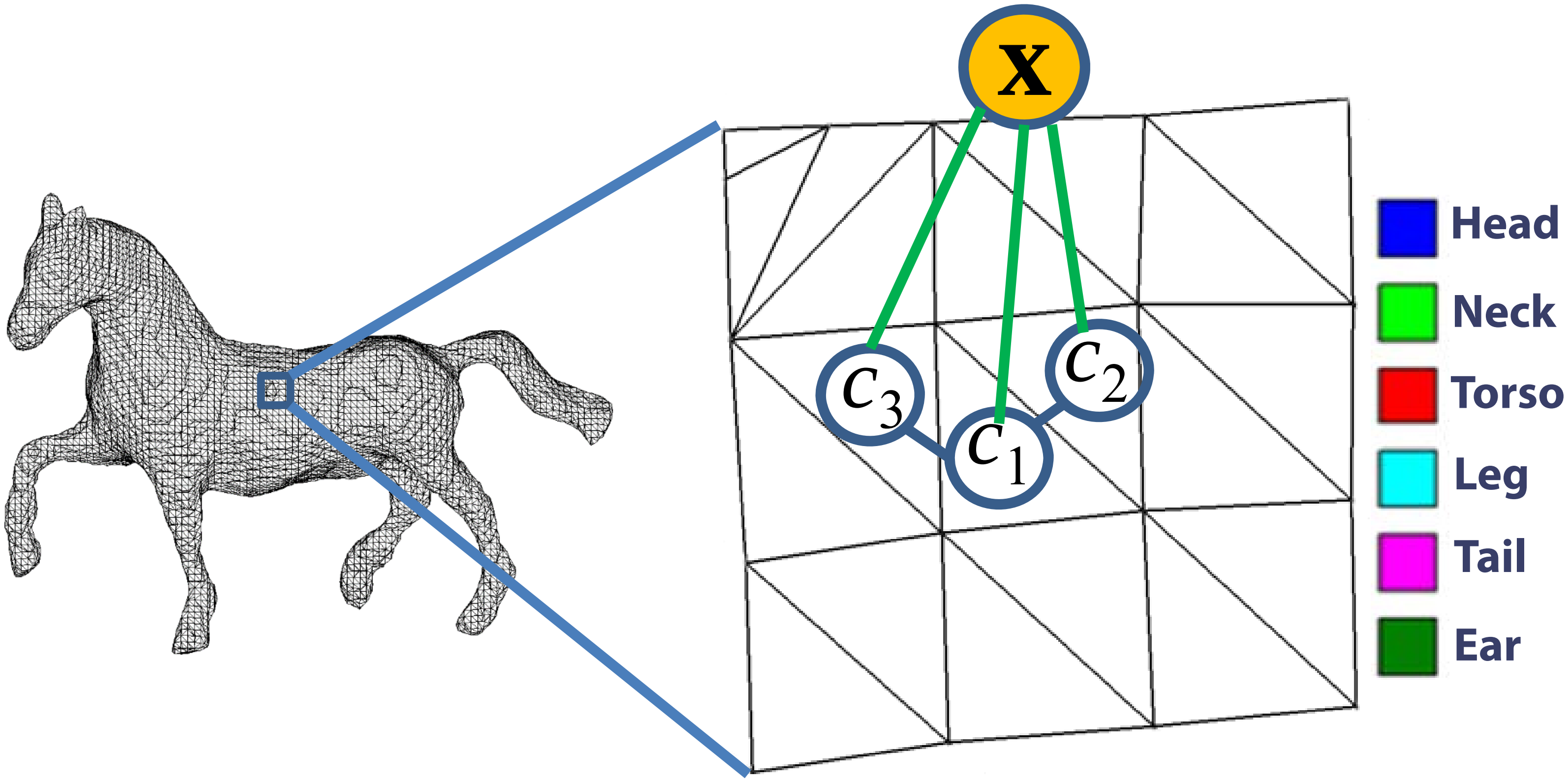
$$c_1, c_2, c_3 \in C$$

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

Feature vector $\mathbf{x}$



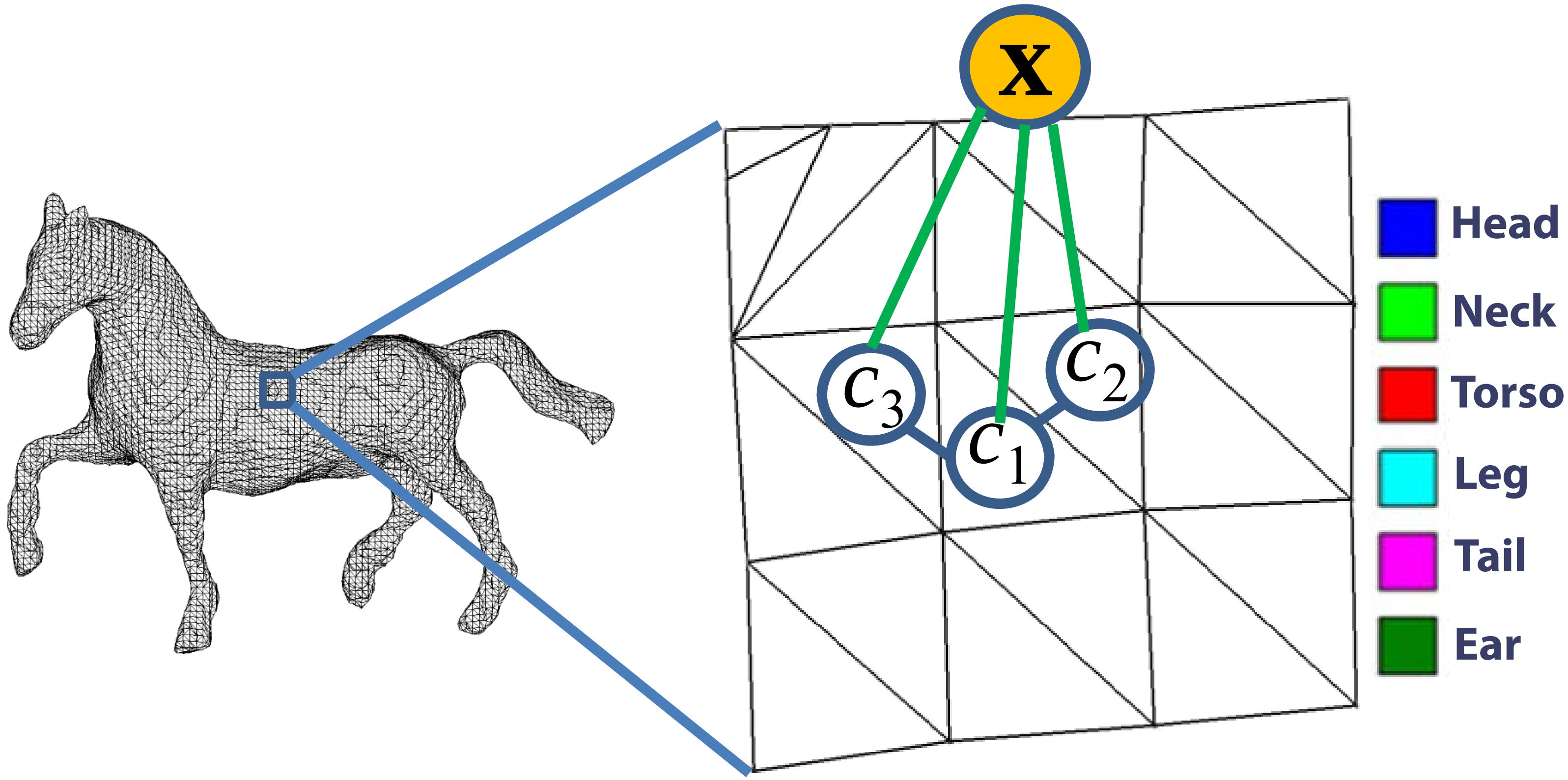
Labeling problem statement



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

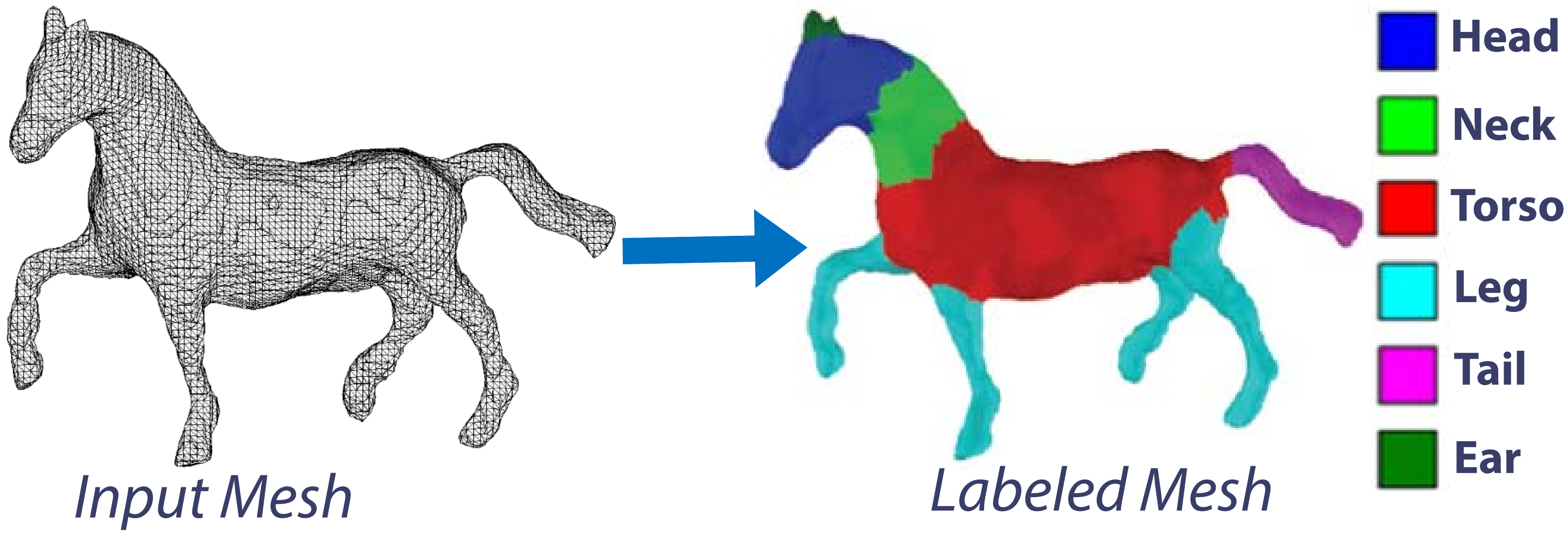
$$C = \{\, \textit{head}, \textit{neck}, \textit{torso}, \textit{leg}, \textit{tail}, \textit{ear} \,\}$$

Labeling problem statement



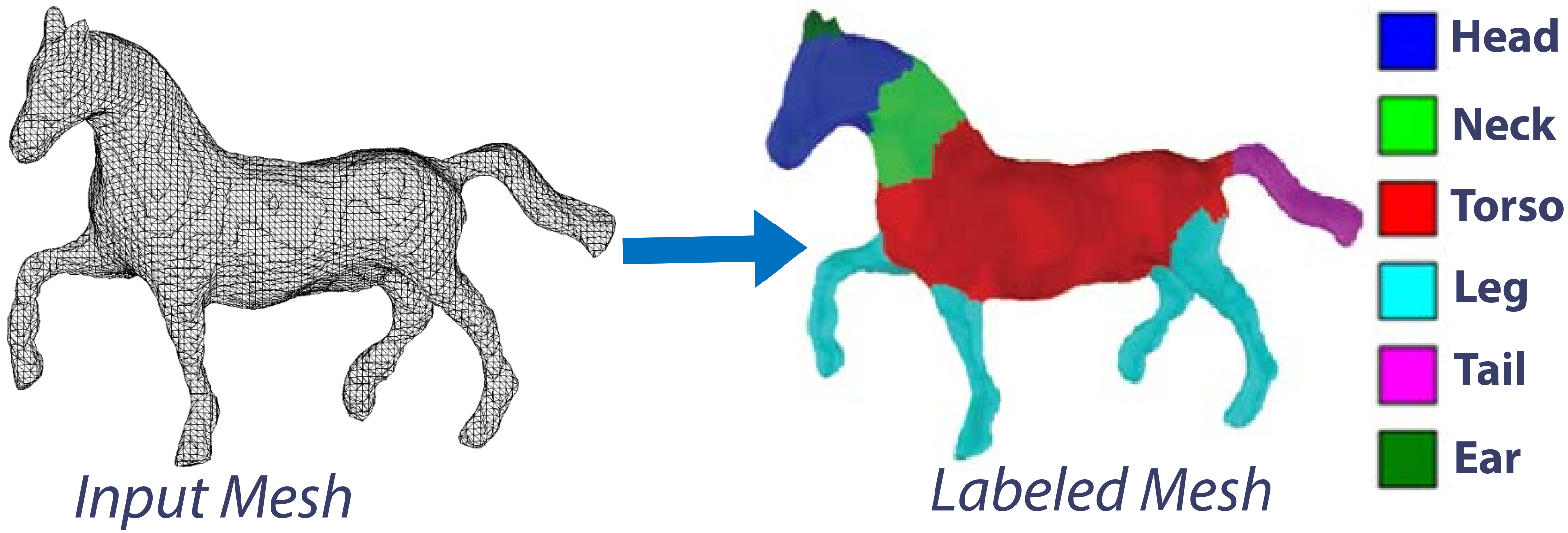
model $P(c_1, c_2, \ldots, c_n \mid \mathbf{x})$

Conditional random field for labeling



$$P(c_1, c_2, \ldots, c_n \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \underbrace{\boxed{\prod_{i=1..n} P(c_i \mid x_i)}}_{\text{Unary term}} \prod_{i,j} P(c_i, c_j \mid x_{ij})$$

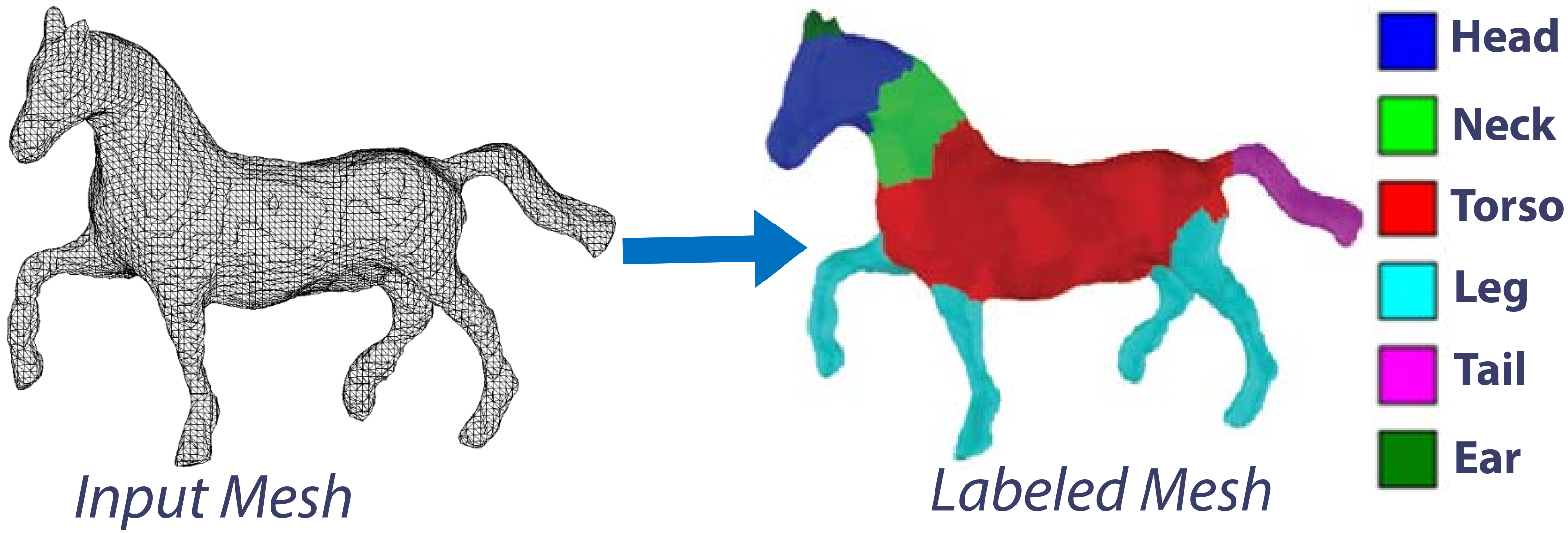
Conditional random field for labeling



$$P(c_1, c_2, \ldots, c_n \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{i=1..n} P(c_i \mid \boxed{x_i}) \prod_{i,j} P(c_i, c_j \mid x_{ij})$$

Face features

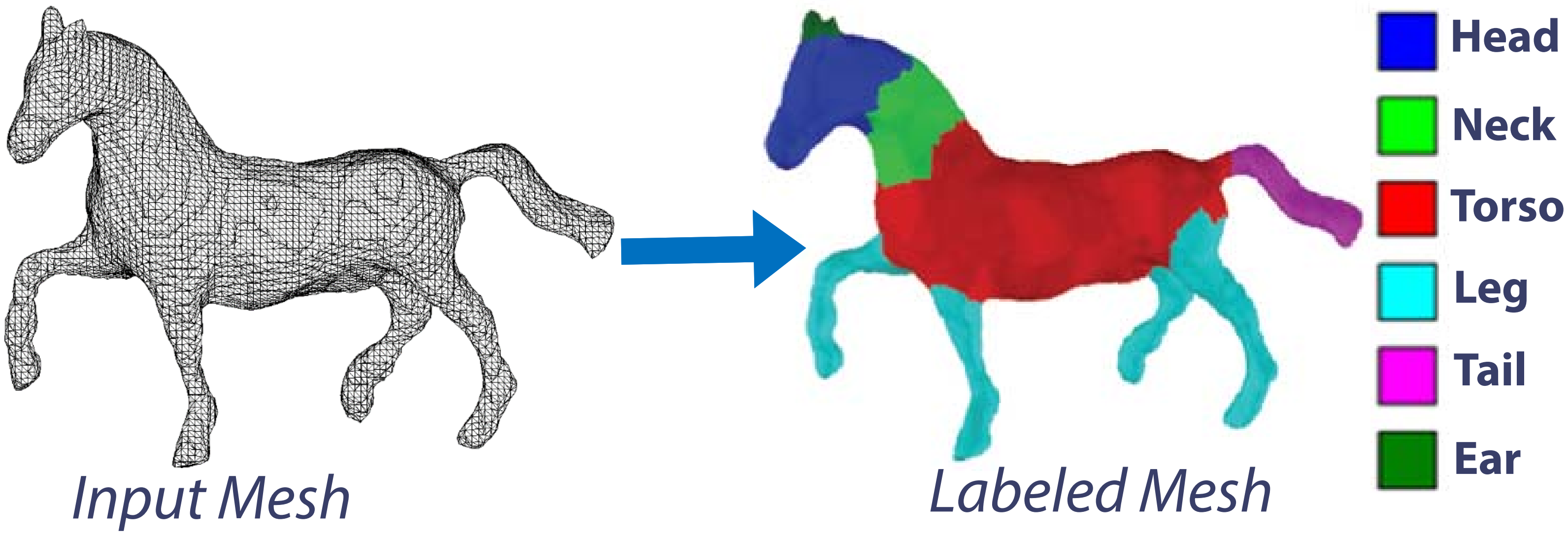
Conditional random field for labeling



$$P(c_1, c_2, \ldots, c_n \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{i=1..n} P(c_i \mid x_i) \boxed{\prod_{i,j} P(c_i, c_j \mid x_{ij})}$$

Pairwise term

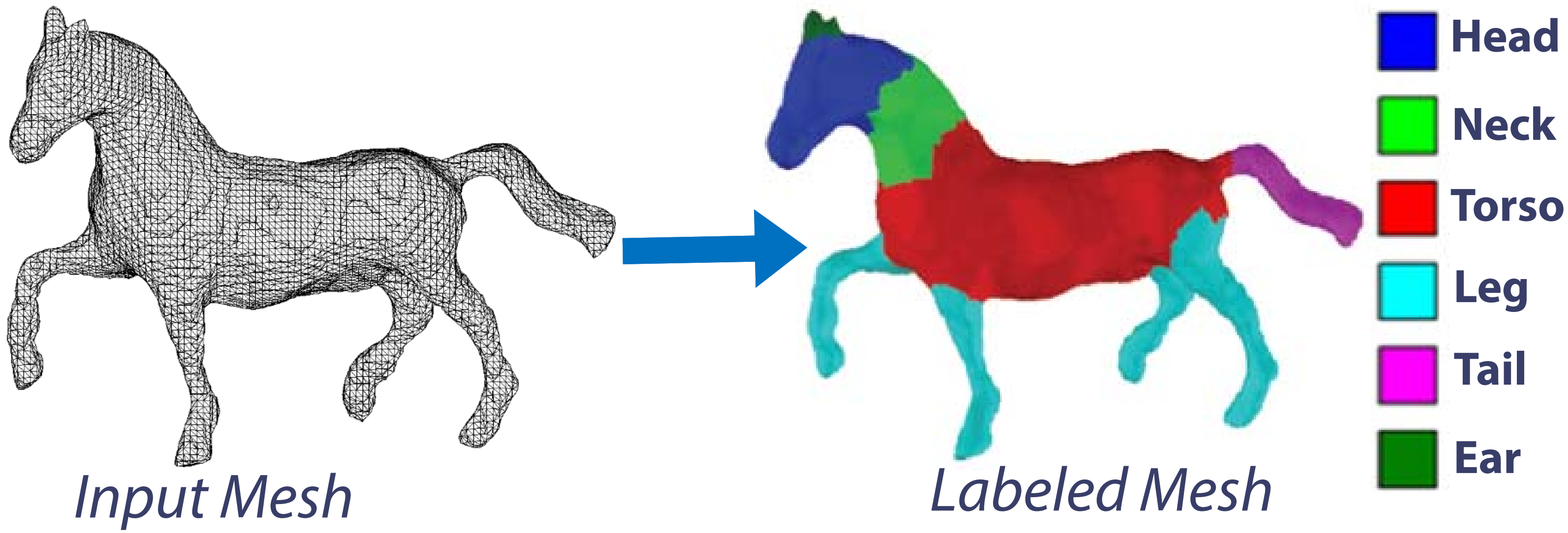
Conditional random field for labeling



$$P(c_1, c_2, \ldots, c_n \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{i=1..n} P(c_i \mid x_i) \prod_{i,j} P(c_i, c_j \mid \boxed{x_{ij}})$$

Pairwise features

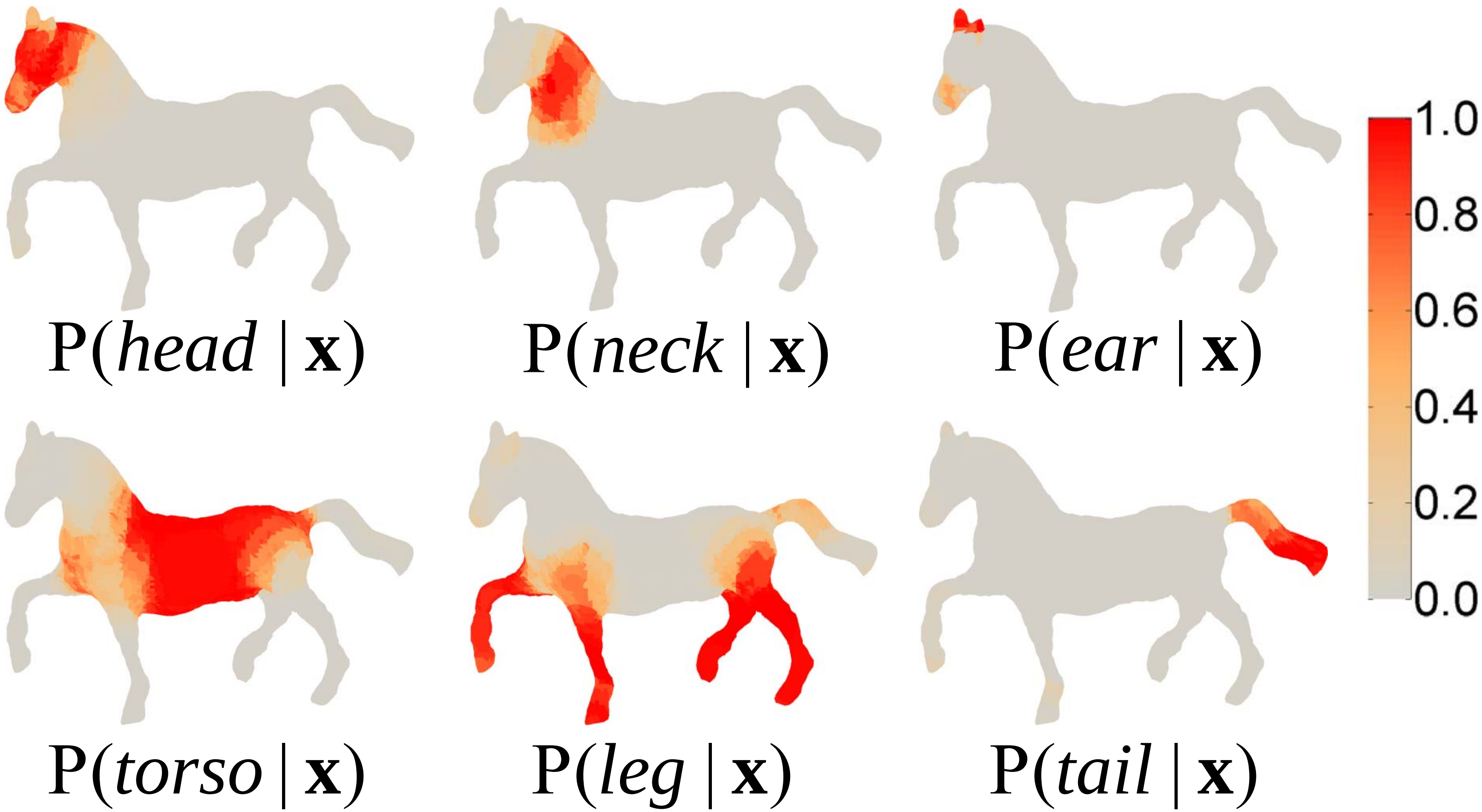
Conditional random field for labeling



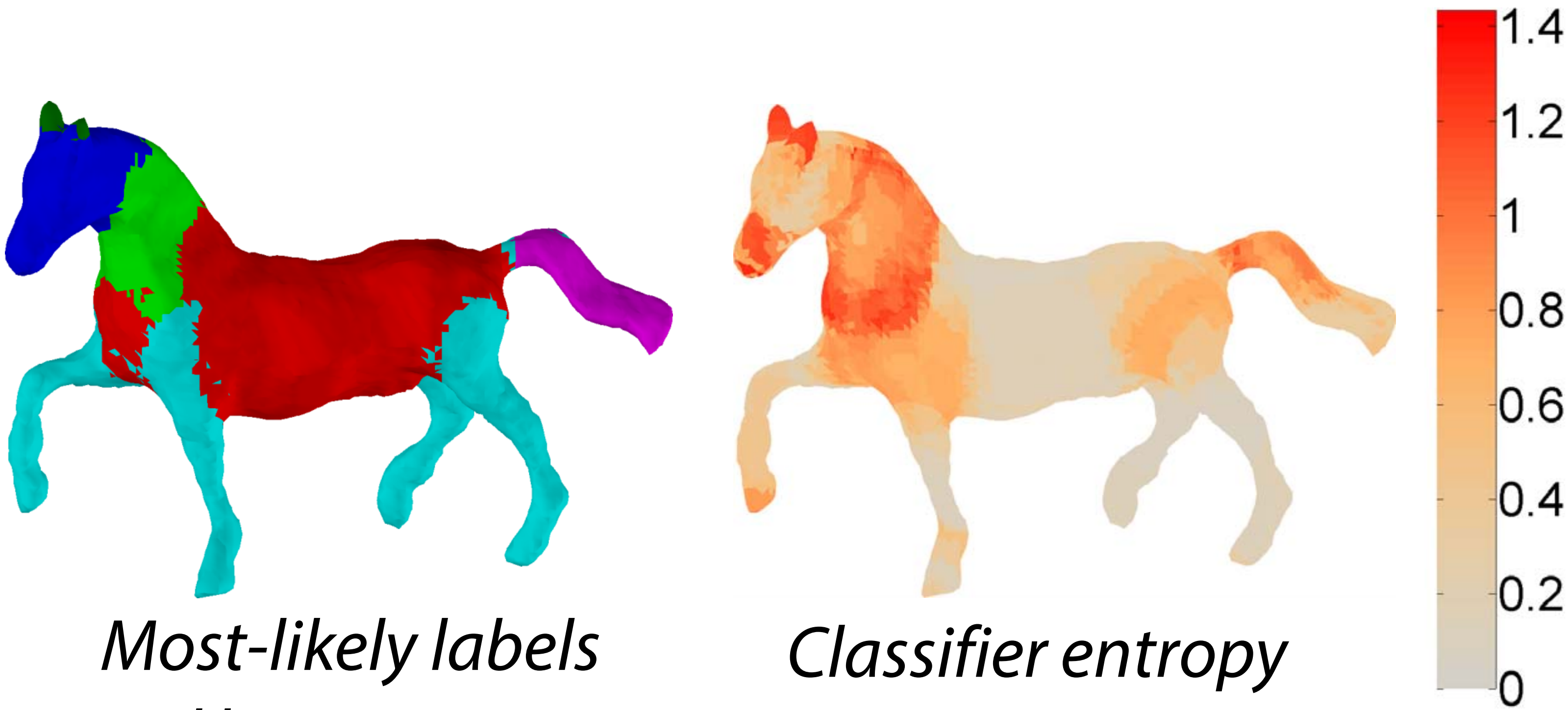
$$P(c_1, c_2, \ldots, c_n \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \underbrace{\boxed{\prod_{i=1..n} P(c_i \mid x_i)}}_{\textit{Unary term}} \prod_{i,j} P(c_i, c_j \mid x_{ij})$$

Unary term

$$P(c_i \mid x_i)$$



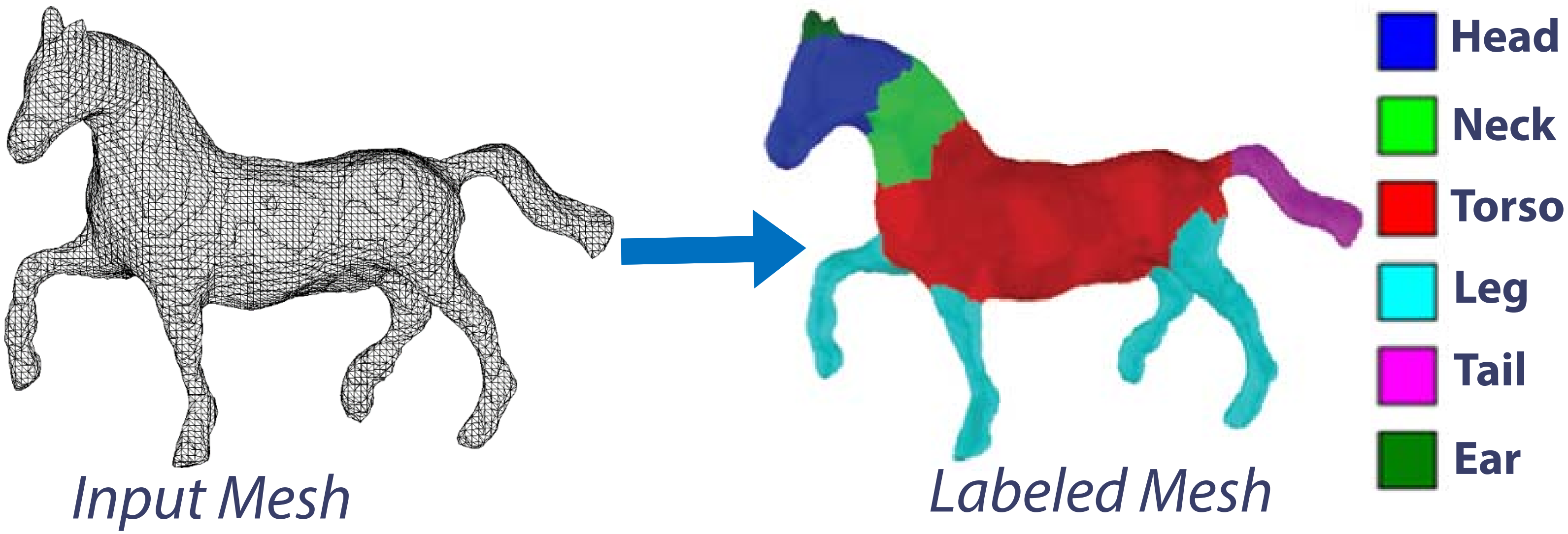
Unary term



Most-likely labels
Unary term

Classifier entropy

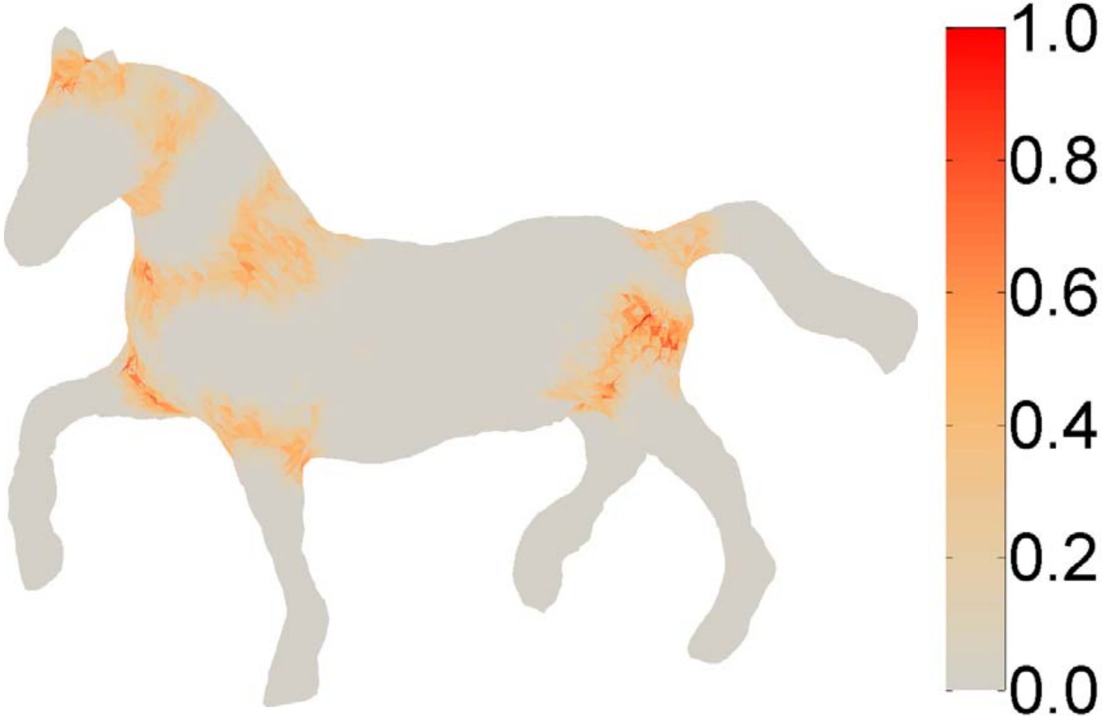
Conditional Random Field for labeling



$$P(c_1, c_2, \ldots, c_n \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{i=1..n} P(c_i \mid x_i) \underbrace{\boxed{\prod_{i,j} P(c_i, c_j \mid x_{ij})}}_{\text{Pairwise term}}$$

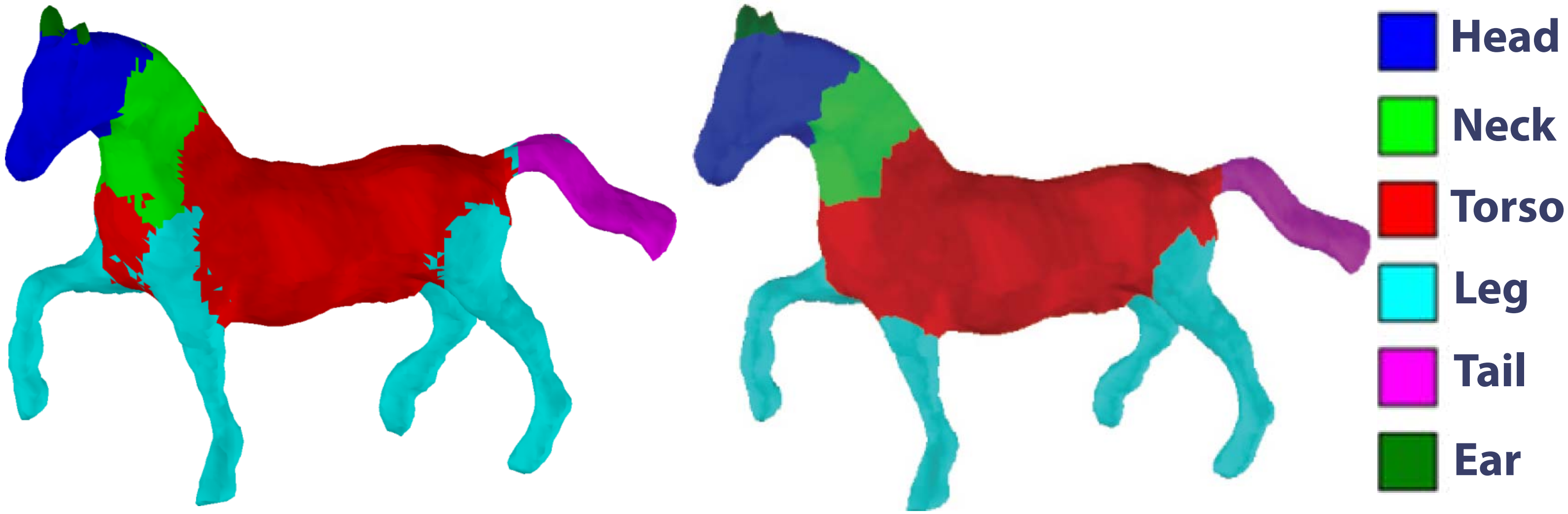
Pairwise Term

$$P(c \neq c' \mid x_{ij})\, L(c, c')$$



Maximum A Posteriori assignment

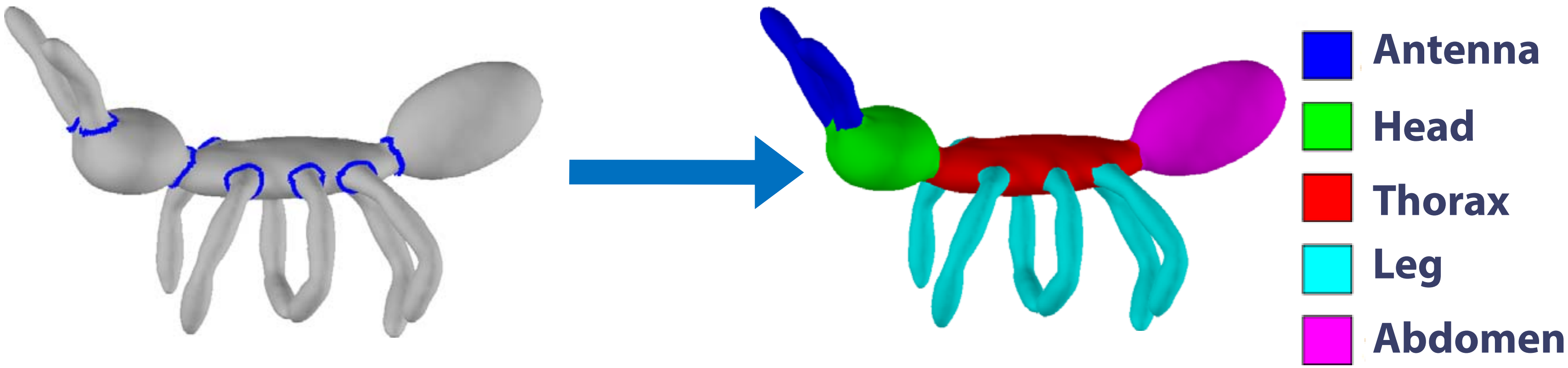
$$\underset{c_1, c_2, \ldots, c_n}{\arg\max}\, P(c_1, c_2, \ldots, c_n \mid \mathbf{x})$$



Unary term classifier *MAP estimate*

Dataset used in experiments

We label 380 meshes (19 categories) from the Princeton Segmentation Benchmark



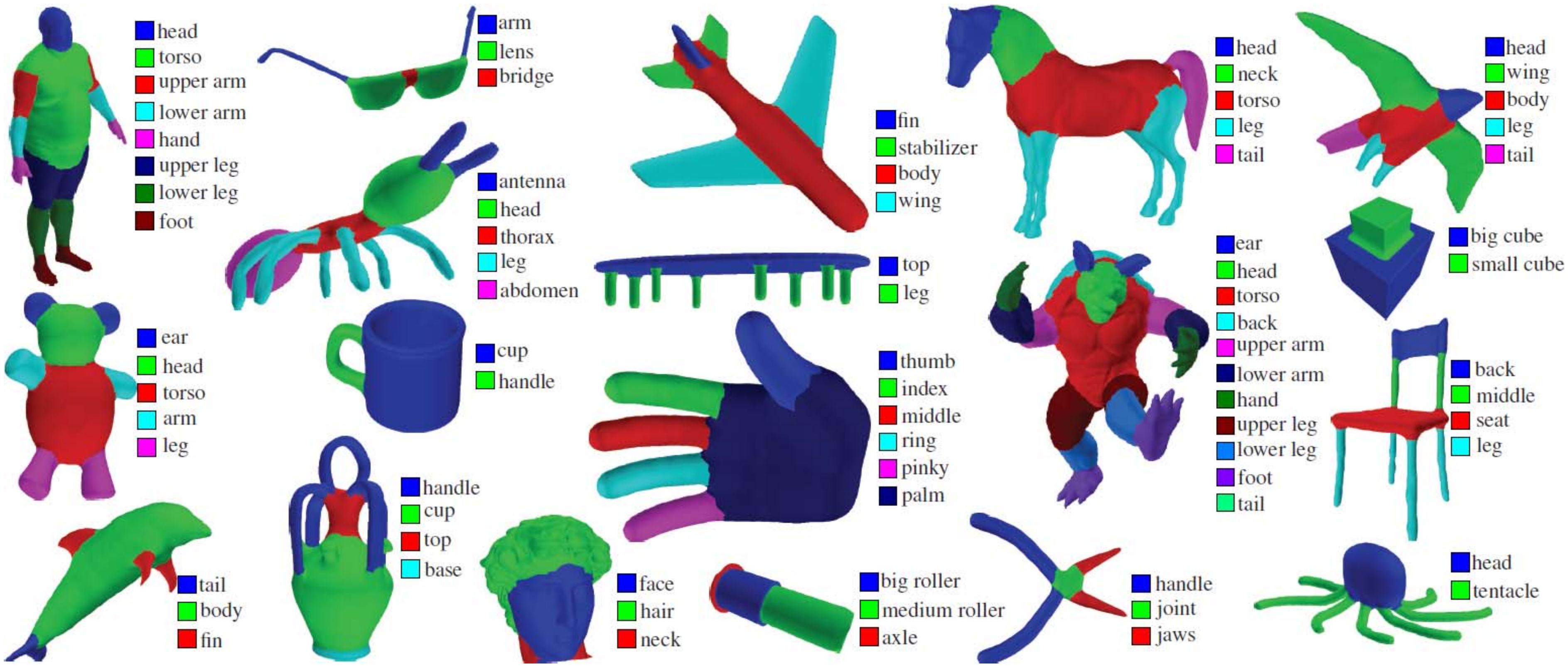
[Chen et al. 2009]

Quantitative Evaluation

<u>Segmentation</u>

- Our result: **9.5%** Rand Index error
- **Outperforms all prior work:**
  - **15%** Randomized Cuts **[Golovinskiy and Funkhouser 08]**
  - **17%** Normalized Cuts **[Golovinskiy and Funkhouser 08]**
  - **17.5%** Shape Diameter **[Shapira et al. 08]**
  - **21%** Core Extraction **[Katz et al. 05]**
  - **21%** Fitting Primitives **[Attene et al. 06]**
  - **21.5%** Random Walks **[Lai et al. 08]**
  - **21%** Intrinsic Symmetry **[Solomon et al. 11]**

Labeling results



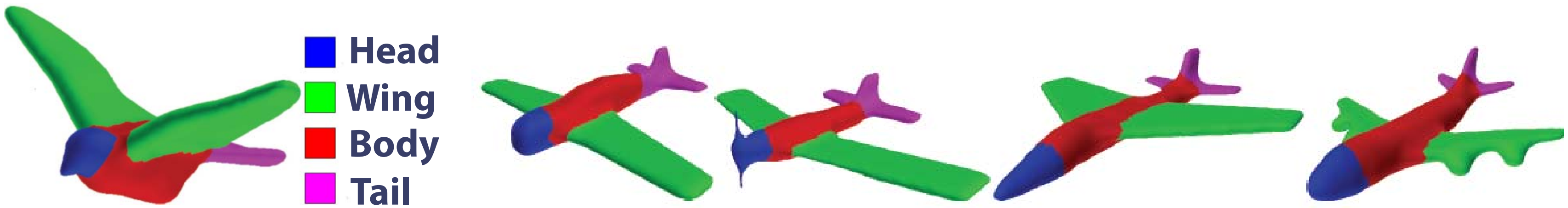
Summary

Use prior knowledge for shape segmentation and labeling

Based on a probabilistic model learned from examples

Significant improvements over the state-of-the-art

Generalization across categories:



Outline

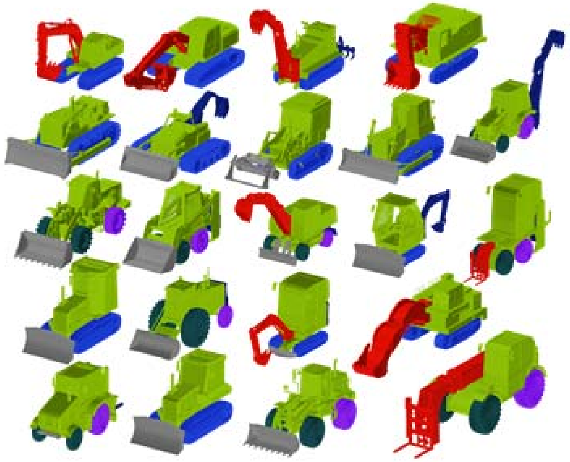
1. Learning 3D shape segmentation and labeling

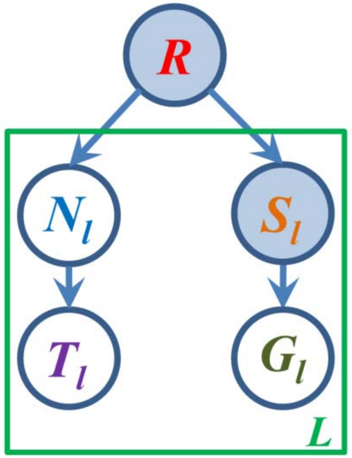
1. A generative model of shapes
   [Kalogerakis et al., SIGGRAPH 2012]

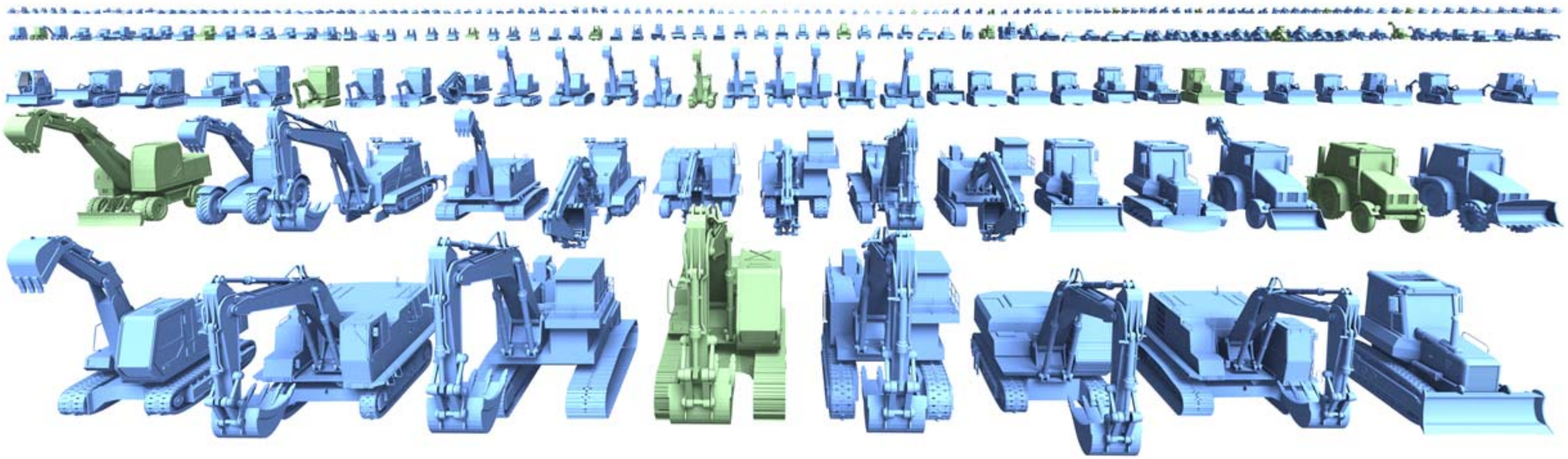
2. Other ML applications to graphics and vision

3. Future work

Goal: generative model of shape



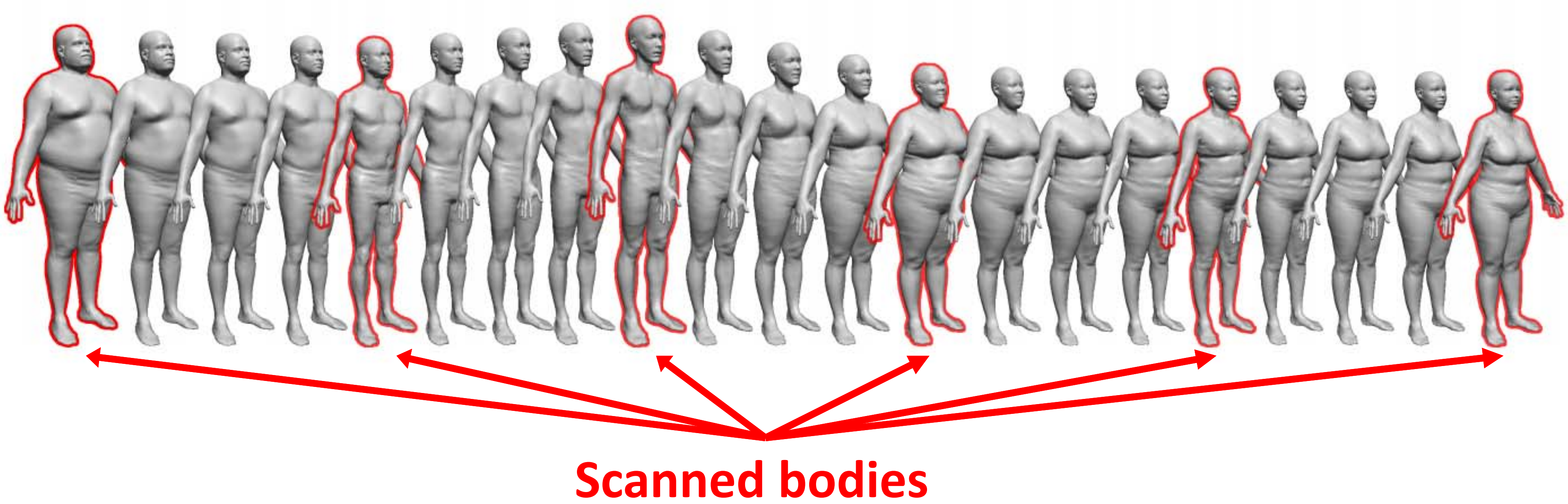




Related work: generative models of bodies & faces

Works on relatively simple shapes with fixed structure

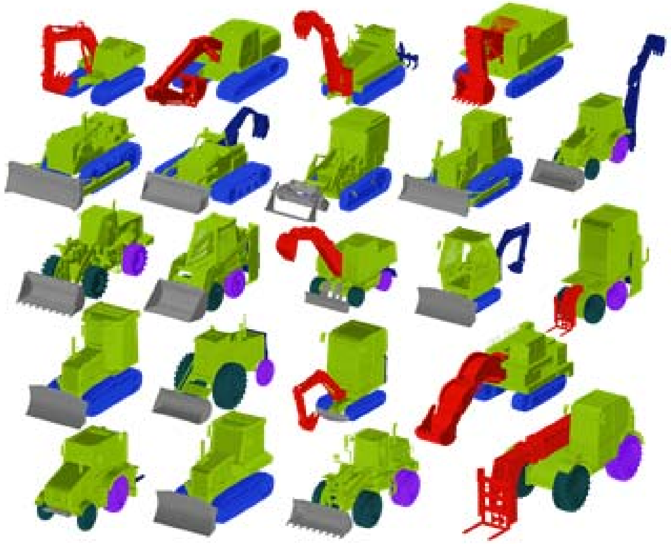
Based on dense correspondences between input shapes



[Allen et al. SIGGRAPH 2003]

Learning shape structure

We want to model attributes related to shape structure

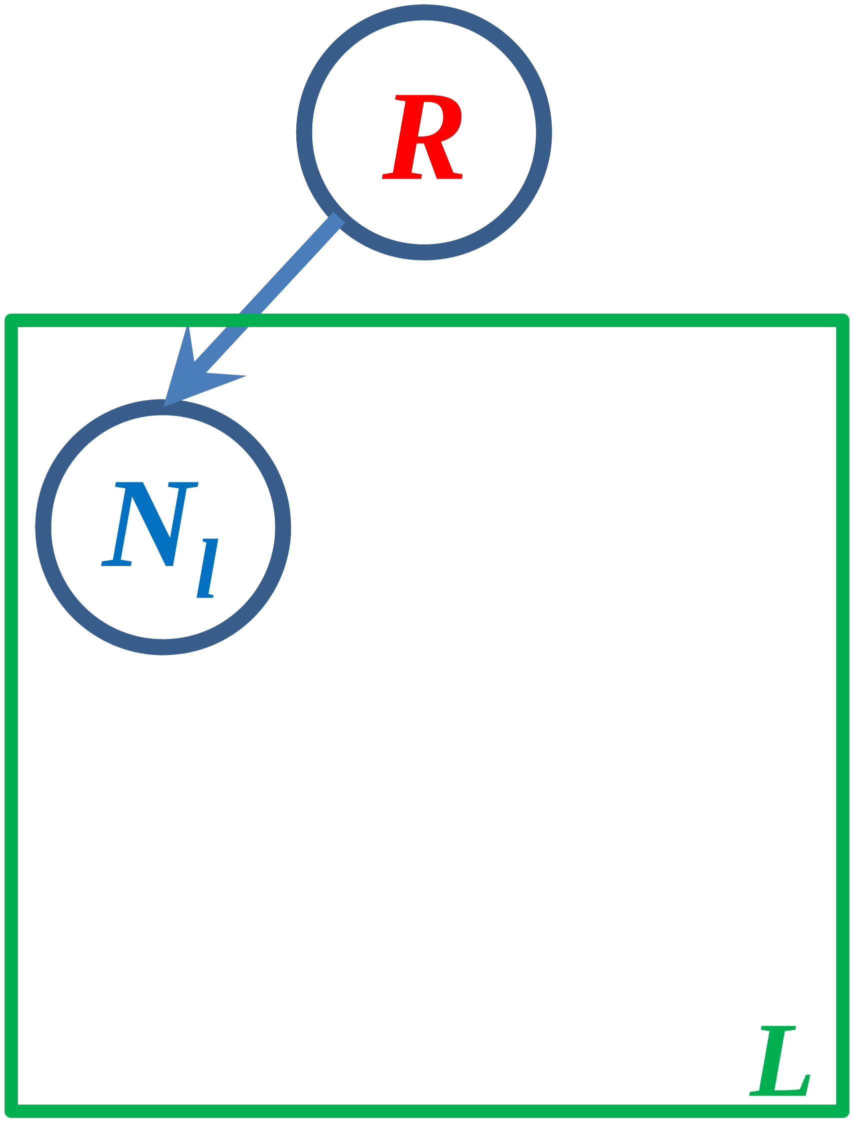


Shape styles
Component styles
Number of components
Component geometry
Component placement

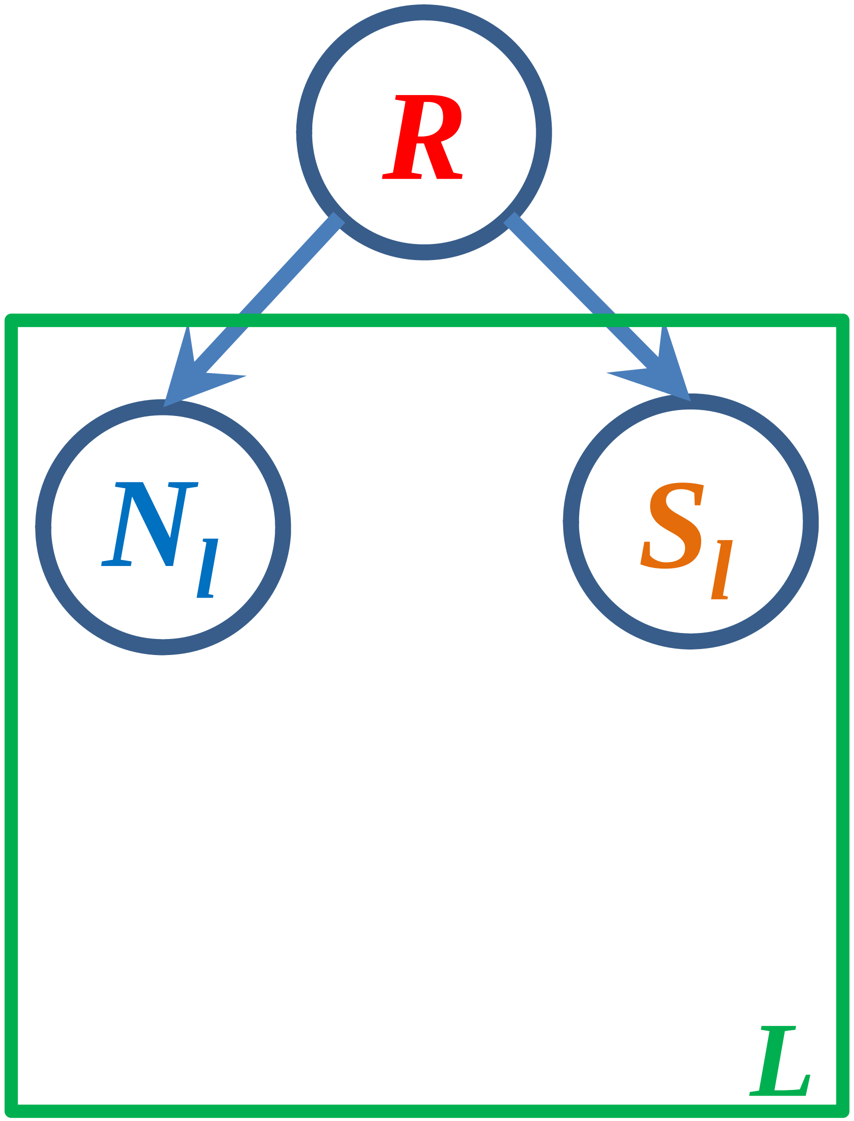
model $P(R, \{S_l\}, \{N_l\}, \{G_l\}, \{T_l\})$

$\mathrm{P}(R)$

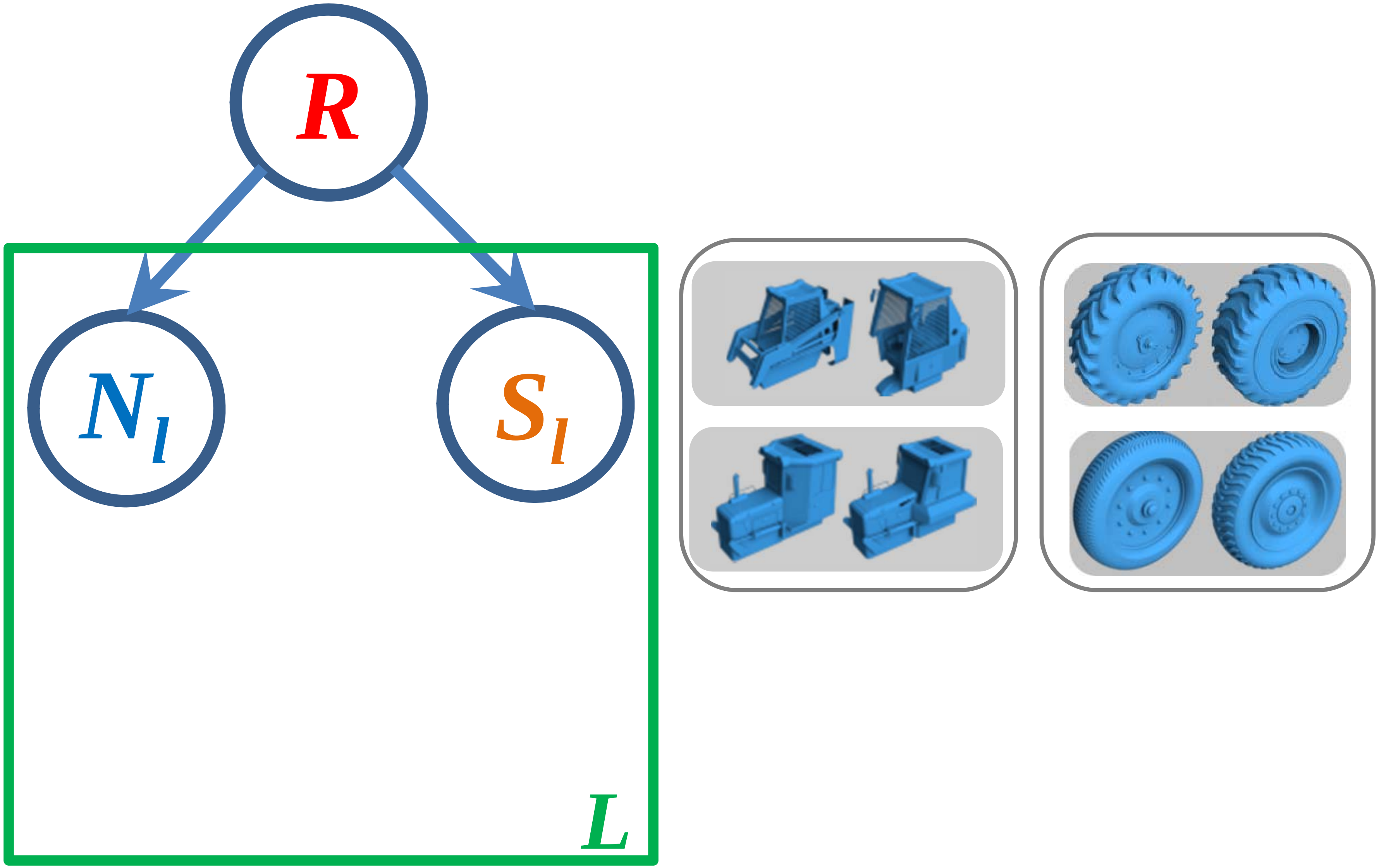
$P(R)$



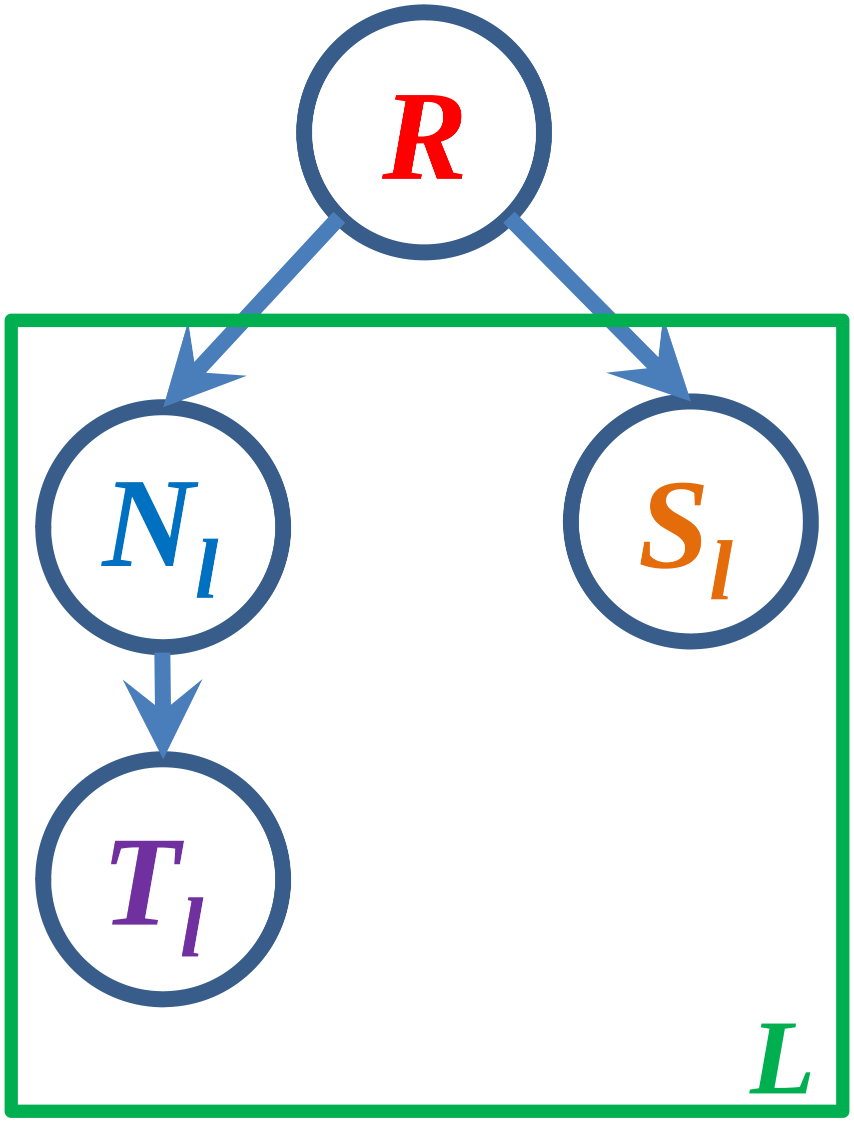
$$P(R) \prod_{l \in L} \left[P(N_l \mid R) \right]$$



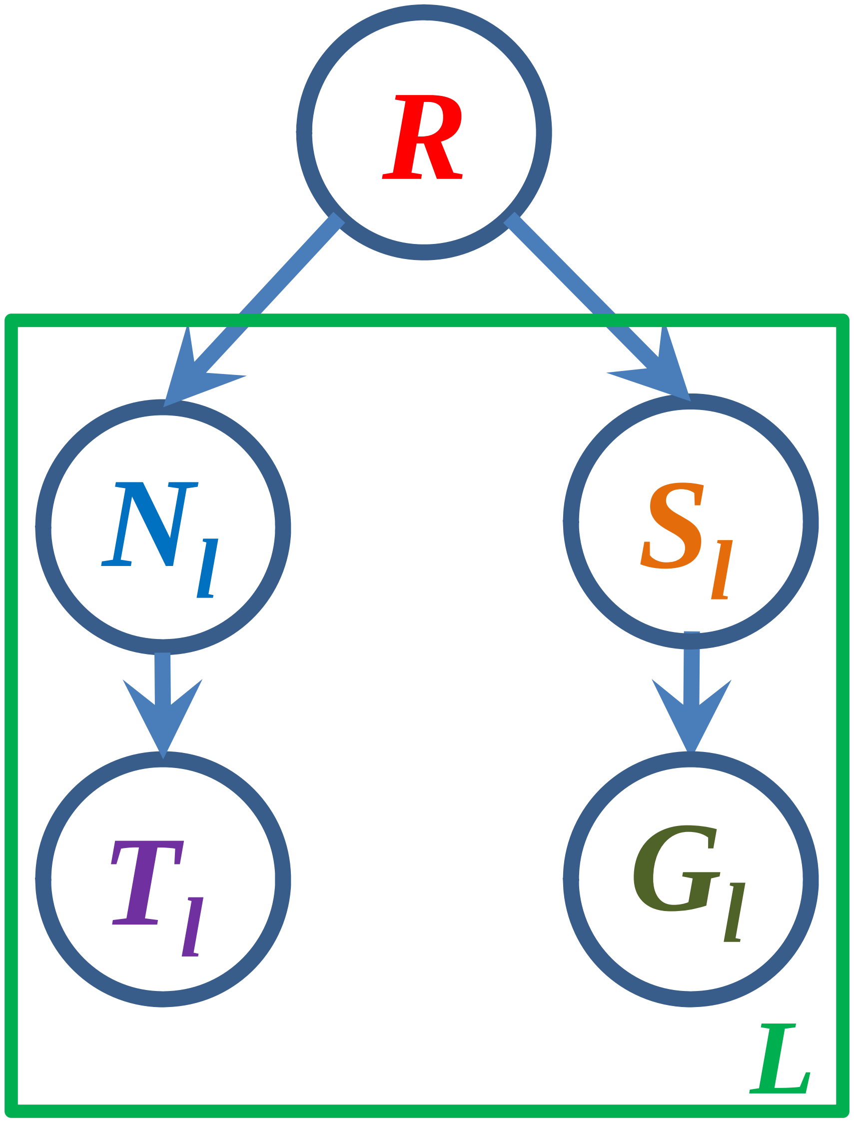
$$P(R)\prod_{l\in L}\left[P(N_l \mid R)\,P(S_l \mid R)\right]$$



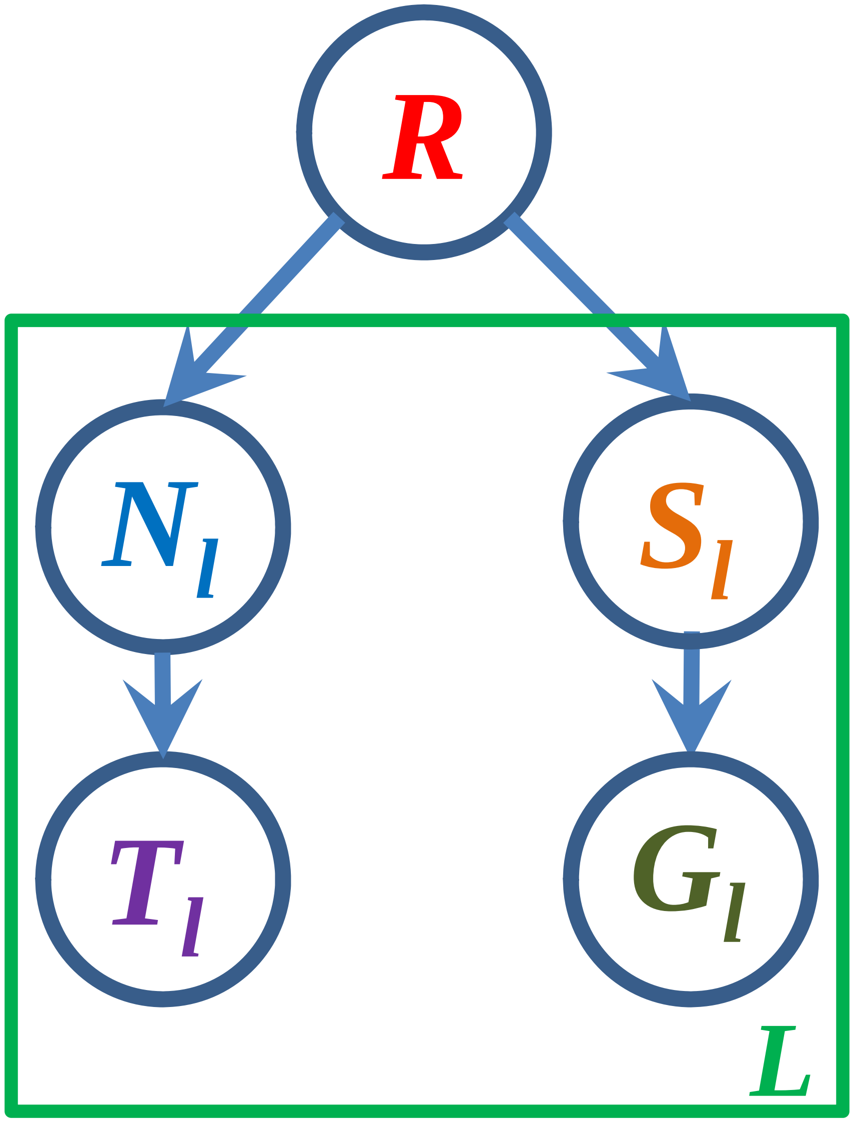
$$P(R)\prod_{l \in L}\left[P(N_l \mid R)\,P(S_l \mid R)\right]$$

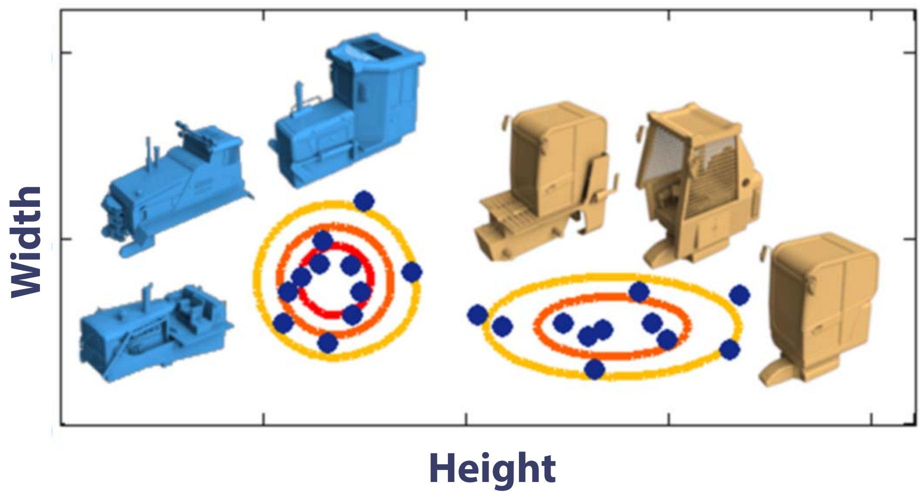


$$P(R)\prod_{l\in L}[P(N_l \mid R)\,P(S_l \mid R)\,P(T_l \mid N_l)]$$

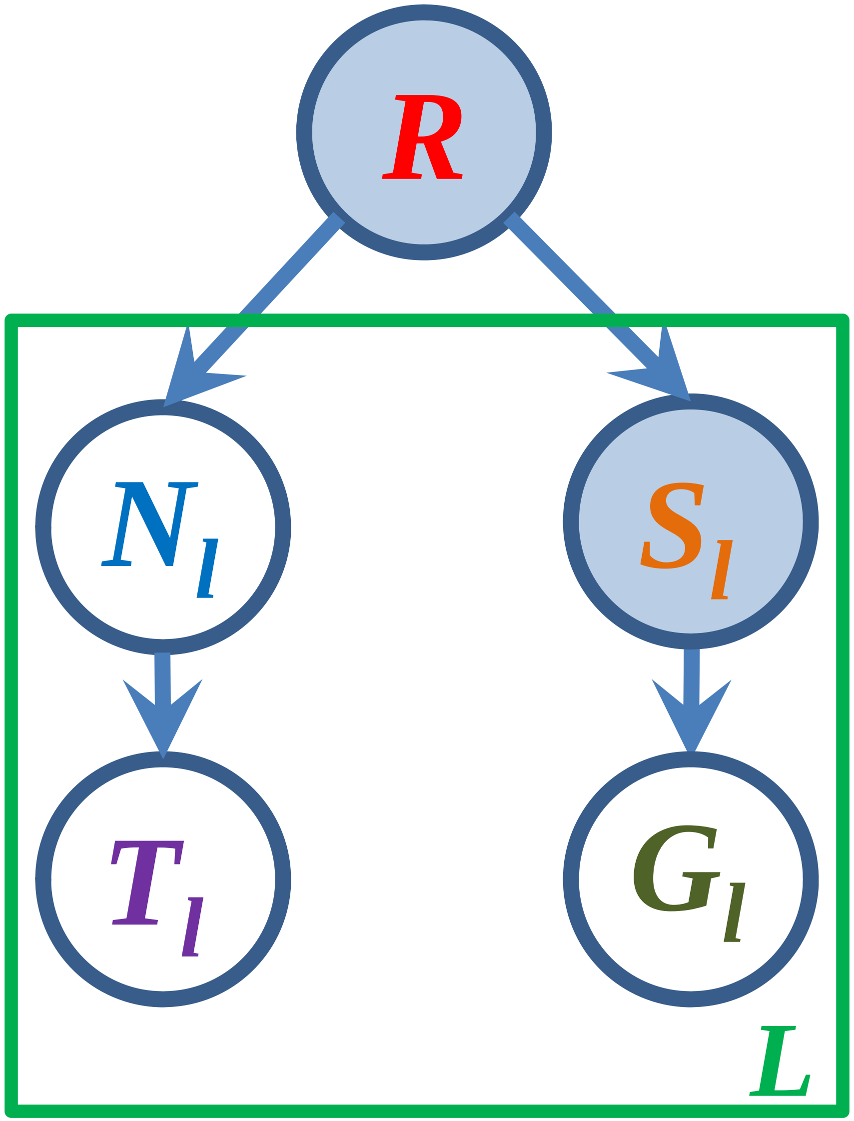


$$P(R)\prod_{l\in L}\left[P(N_l \mid R)\,P(S_l \mid R)\,P(T_l \mid N_l)\,P(G_l \mid S_l)\right]$$





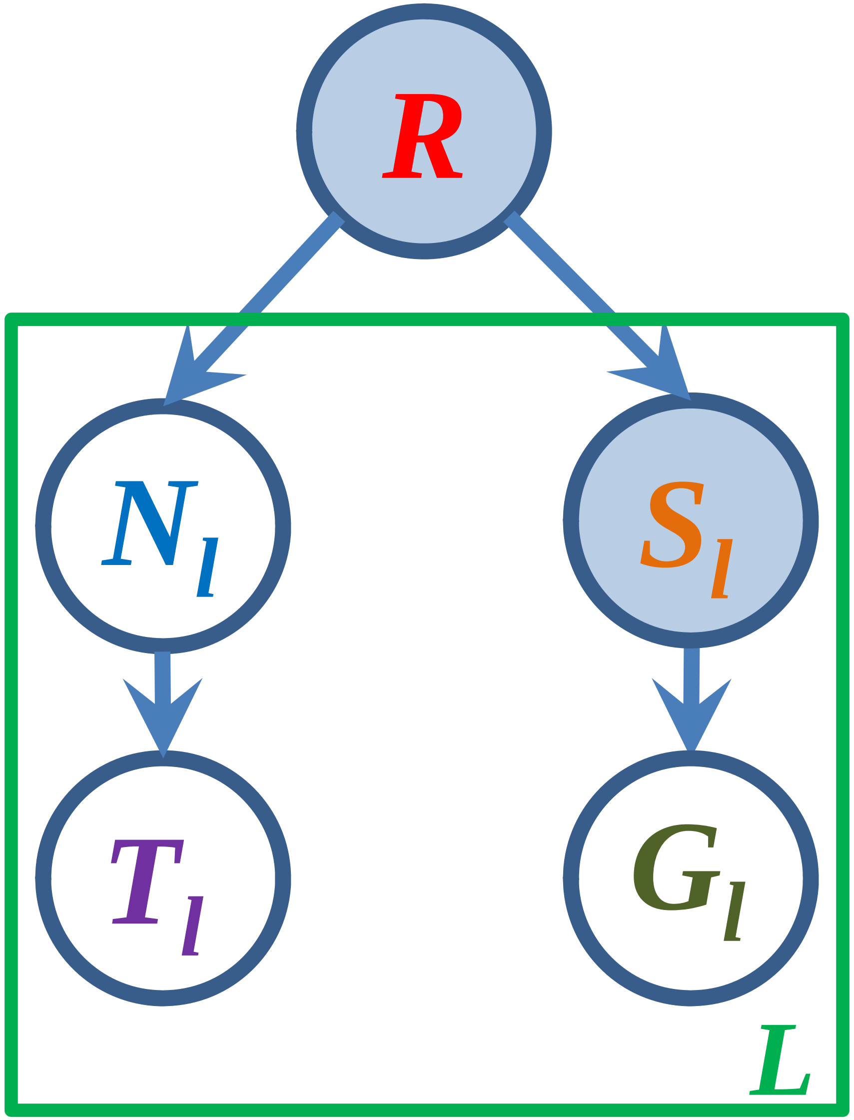
$$P(R)\prod_{l\in L}\left[P(N_l \mid R)\,P(S_l \mid R)\,P(T_l \mid N_l)\,P(G_l \mid S_l)\right]$$

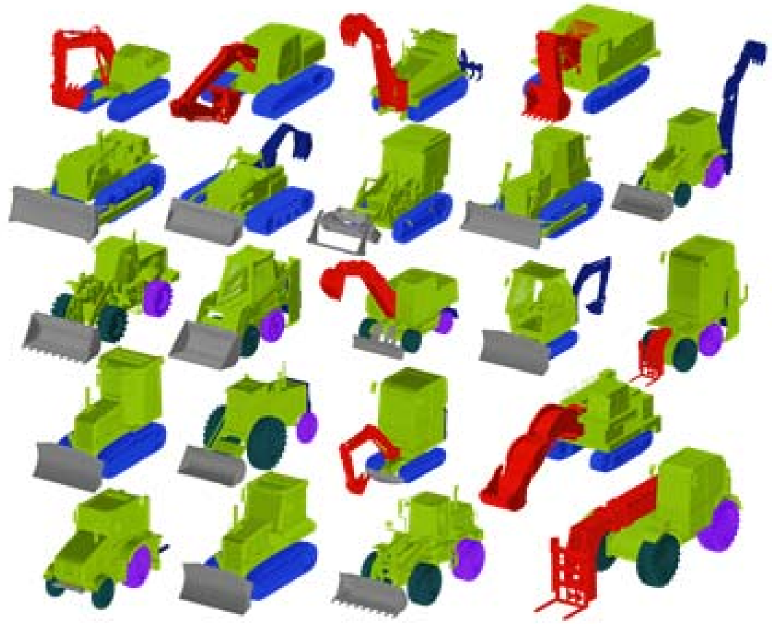


Latent object style

Latent component style

$$P(R) \prod_{l \in L} \left[P(N_l \mid R)\, P(S_l \mid R)\, P(T_l \mid N_l)\, P(G_l \mid S_l) \right]$$





Learn from training data:

latent styles

lateral edges

parameters of CPDs

Learning

Given observed data $\mathbf{O}$, find structure $\mathbf{G}$ that maximizes:

$$P(G \mid \mathbf{O}) = \frac{\boxed{P(\mathbf{O} \mid G)}\boxed{P(G)}}{P(\mathbf{O})}$$

Assuming uniform prior over structures, maximize **marginal likelihood**:

$$P(\mathbf{O} \mid G) = \sum_{R,\mathbf{S}} \int_{\Theta} P(\mathbf{O}, R, \mathbf{S} \mid \Theta, G) P(\Theta \mid G)\, \mathrm{d}\Theta$$

Learning

Given observed data $\mathbf{O}$, find structure $\mathbf{G}$ that maximizes:

$$P(G \mid \mathbf{O}) = \frac{P(\mathbf{O} \mid G)P(G)}{P(\mathbf{O})}$$

Assuming uniform prior over structures, maximize **marginal likelihood**:

$$P(\mathbf{O} \mid G) = \sum_{R,\mathbf{S}} \int_{\Theta} \boxed{P(\mathbf{O}, R, \mathbf{S} \mid \Theta, G)} P(\Theta \mid G)\, \mathrm{d}\Theta$$

Complete likelihood

Learning

Given observed data $\mathbf{O}$, find structure $\mathbf{G}$ that maximizes:

$$P(G \mid \mathbf{O}) = \frac{P(\mathbf{O} \mid G)P(G)}{P(\mathbf{O})}$$

Assuming uniform prior over structures, maximize **marginal likelihood**:

$$P(\mathbf{O} \mid G) = \sum_{R,\mathbf{S}} \int_{\Theta} P(\mathbf{O}, R, \mathbf{S} \mid \Theta, G) \boxed{P(\Theta \mid G)} \mathrm{d}\Theta$$

Parameter priors

Learning

Given observed data $\mathbf{O}$, find structure $\mathbf{G}$ that maximizes:

$$P(G \mid \mathbf{O}) = \frac{P(\mathbf{O} \mid G)P(G)}{P(\mathbf{O})}$$

Assuming uniform prior over structures, maximize **marginal likelihood**:

$$P(\mathbf{O} \mid G) = \boxed{\sum_{R,\mathbf{S}} \int_{\Theta}} P(\mathbf{O}, R, \mathbf{S} \mid \Theta, G) P(\Theta \mid G)\, \mathrm{d}\Theta$$

Marginal likelihood

$$P(\mathbf{O} \mid G) = \boxed{\sum_{R,\mathbf{S}}} \int_{\Theta} P(\mathbf{O}, R, \mathbf{S} \mid \Theta, G) P(\Theta \mid G) \, \mathrm{d}\Theta$$

Summation over all possible assignments to the latent variables

Marginal likelihood

$$P(\mathbf{O} \mid G) = \sum_{R,\mathbf{S}} \int_{\Theta} P(\mathbf{O}, \boxed{R, \mathbf{S}} \mid \Theta, G) P(\Theta \mid G) \, \mathrm{d}\Theta$$

need inference for each data instance

Cheeseman-Stutz score

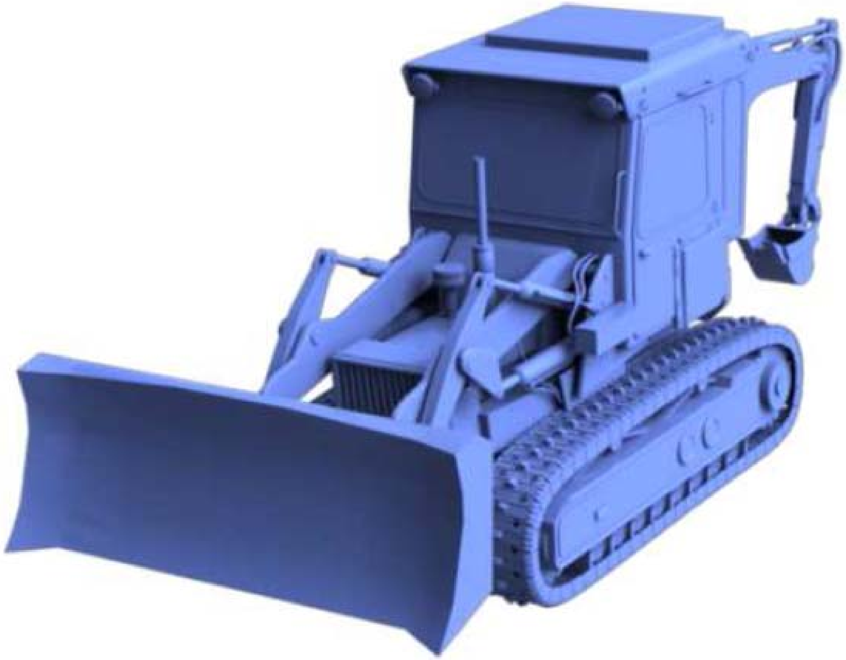
$$P(\mathbf{O} \mid G) \approx \boxed{P(\mathbf{O}^* \mid G)} \cdot \boxed{\frac{P(\mathbf{O} \mid G, \tilde{\Theta}_G)}{P(\mathbf{O}^* \mid G, \tilde{\Theta}_G)}}$$

$\mathbf{O}^*$ is a fictitious dataset composed of training data $\mathbf{O}$ and **approximate statistics for latent variables**

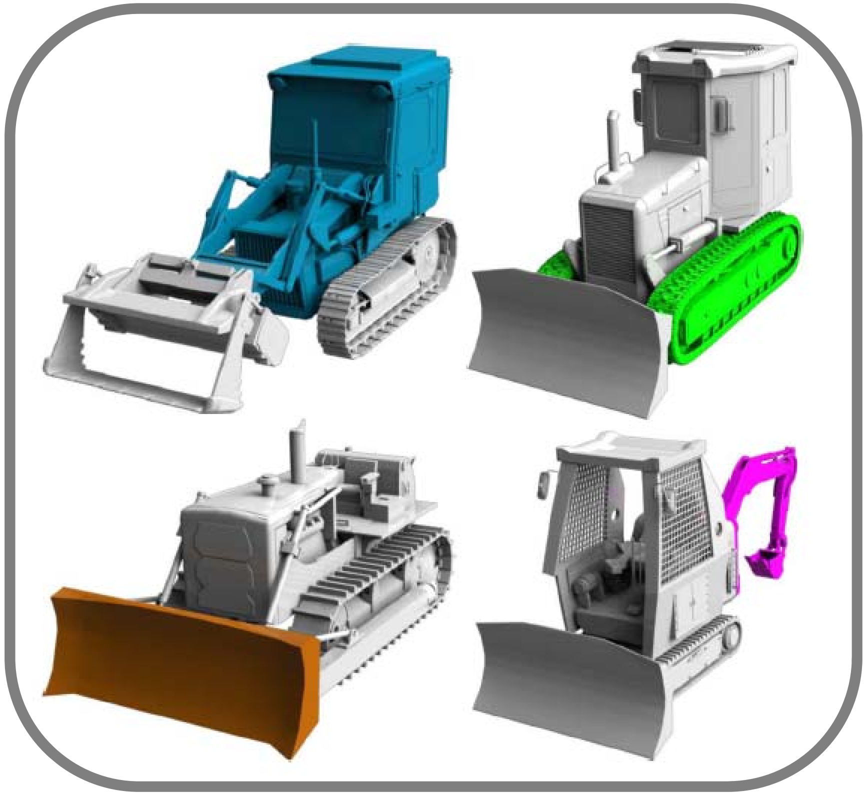
$\tilde{\Theta}_G$ are MAP estimates found by **Expectation-Maximization**

$$\tilde{\Theta}_G = \arg\max_{\Theta} P(\mathbf{O} \mid G, \Theta) P(\Theta \mid G)$$

Shape synthesis



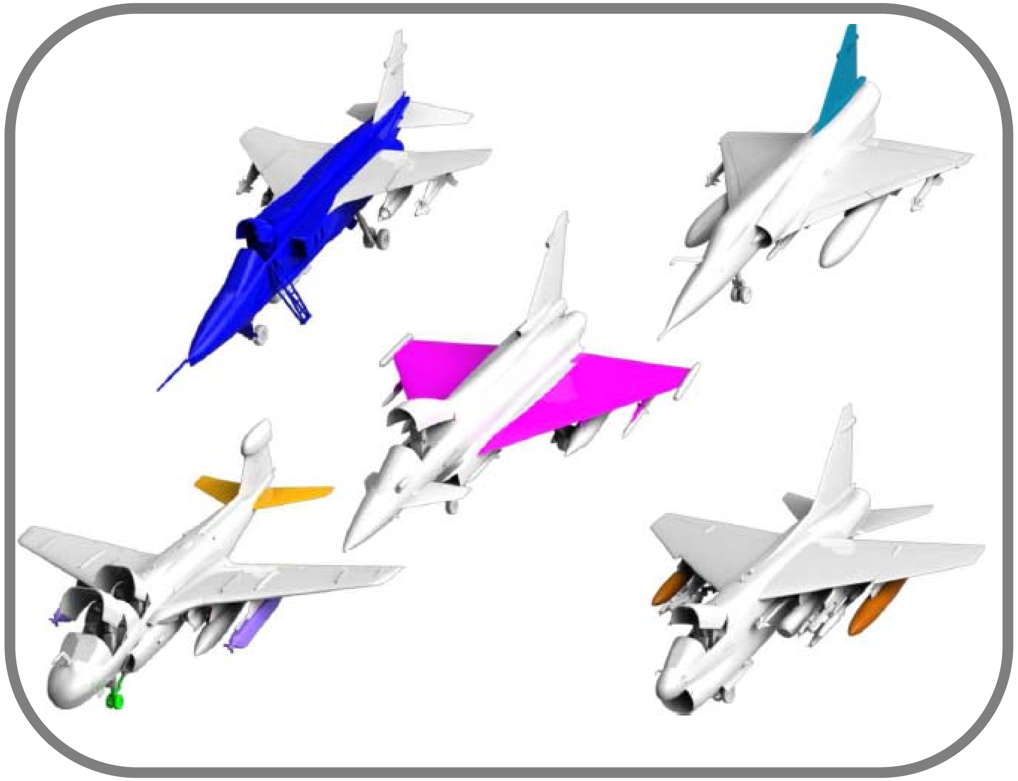
New shape



Source shapes (colored parts are selected for the new shape)

Shape synthesis

New shape

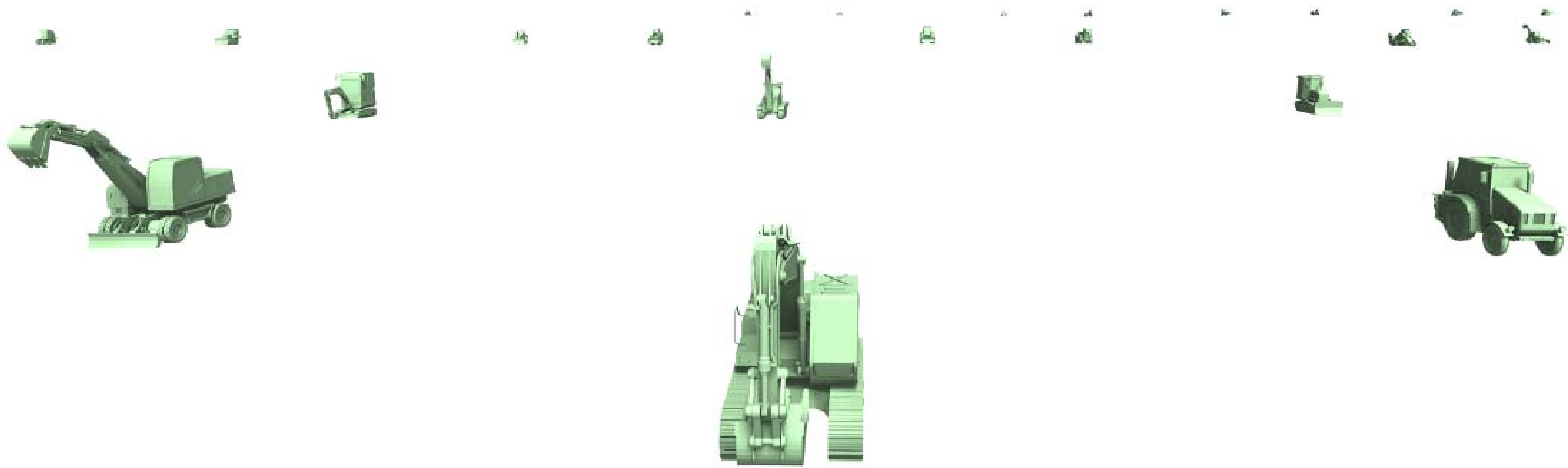


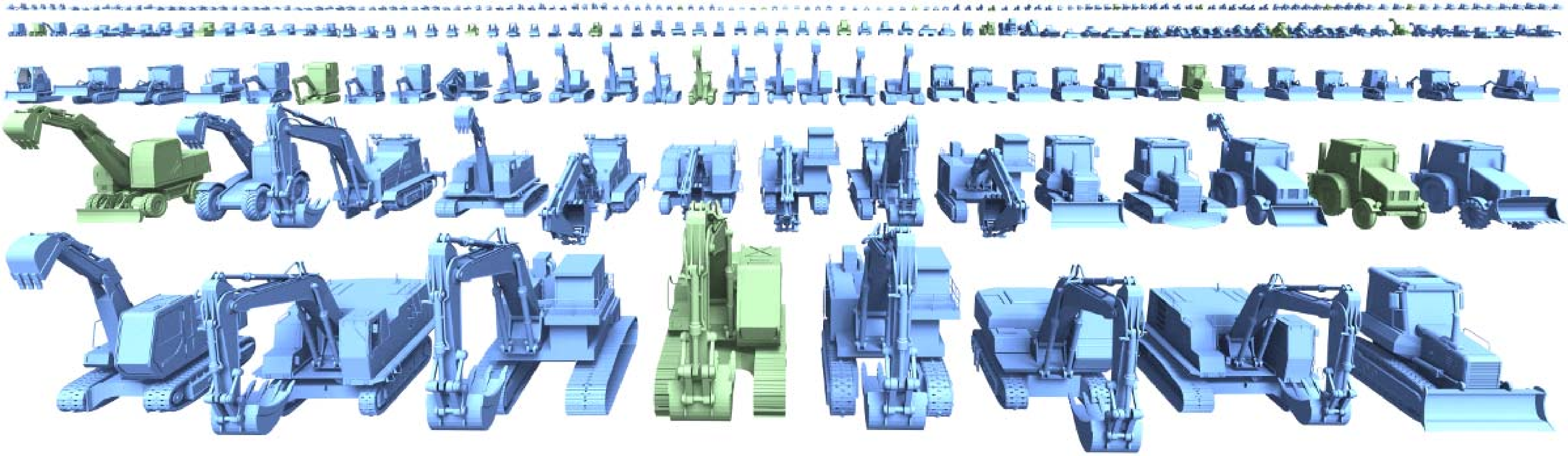
Source shapes
(colored parts are selected for the new shape)

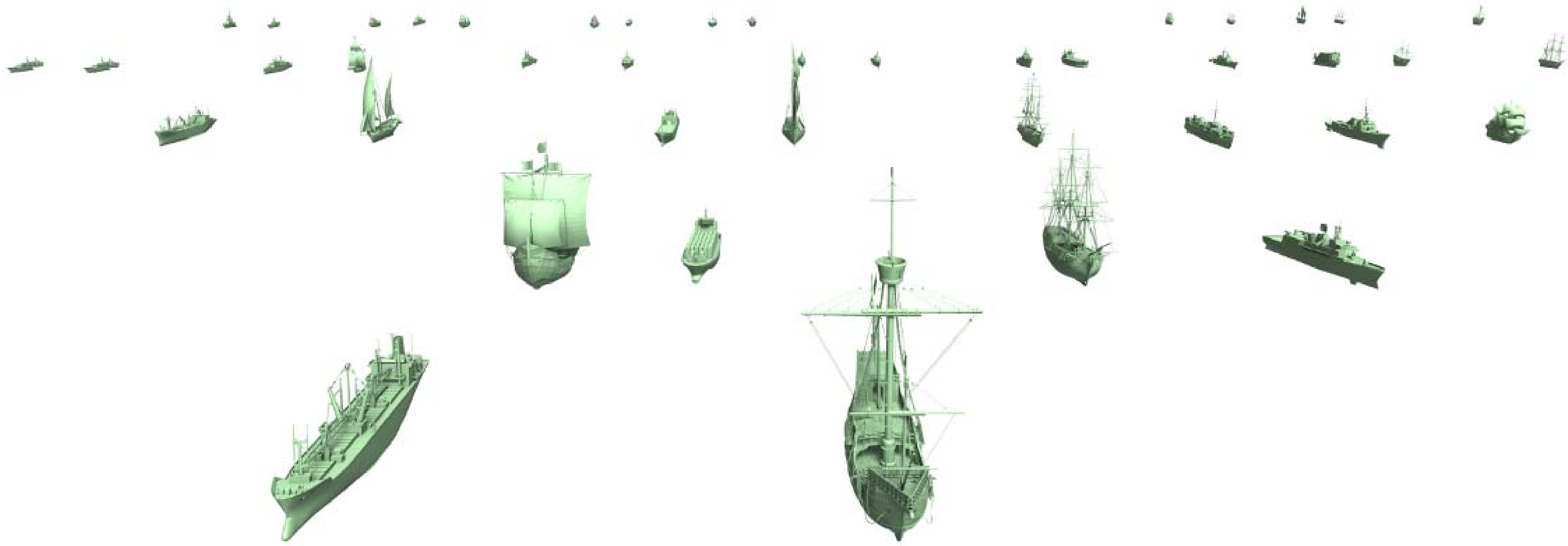
Results of alternative models

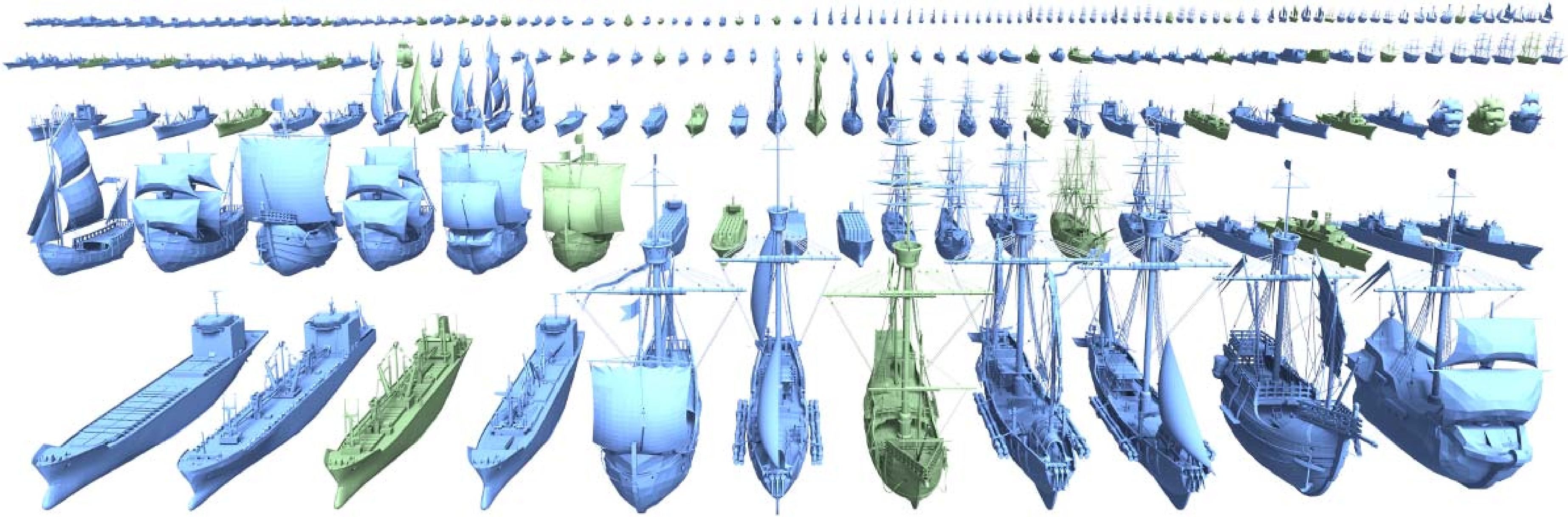
No latent variables

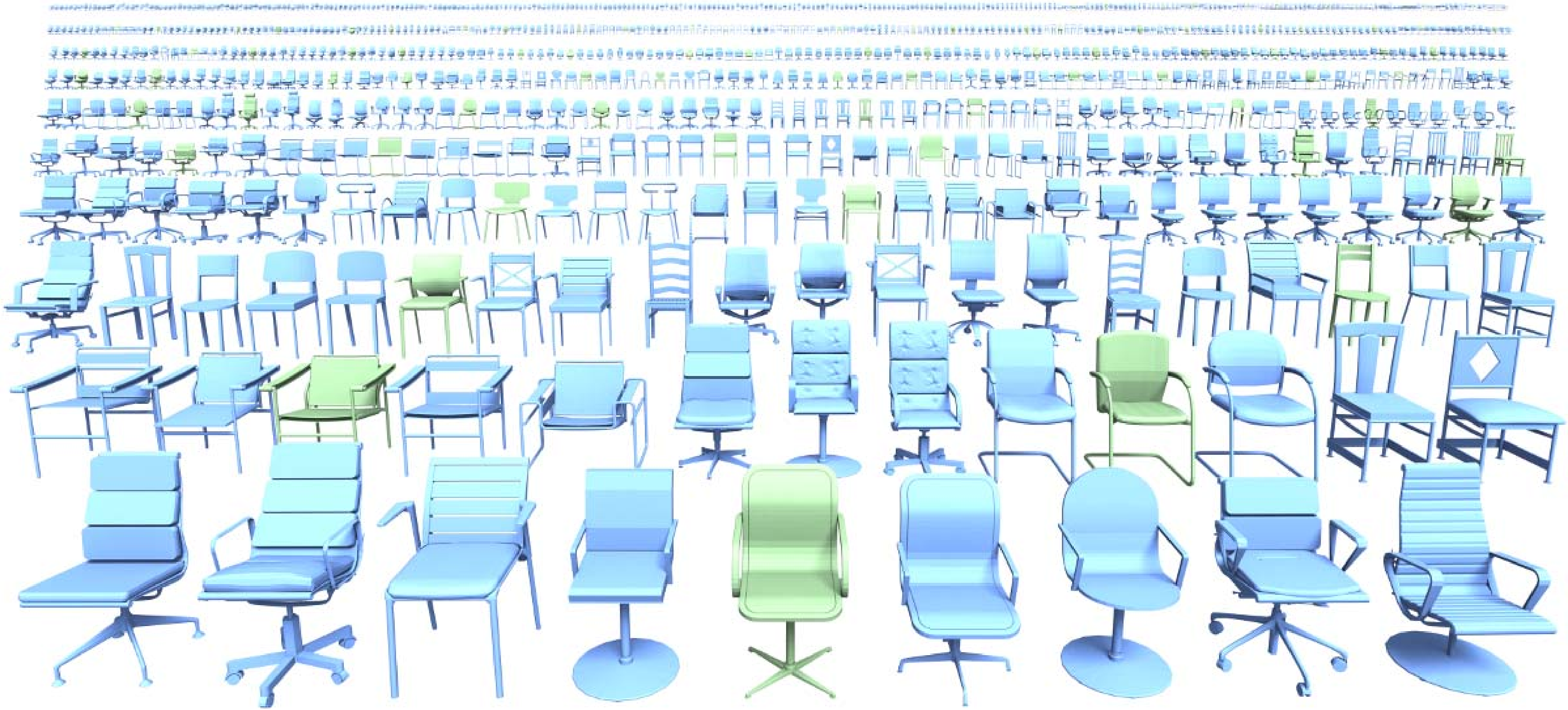
No lateral edges

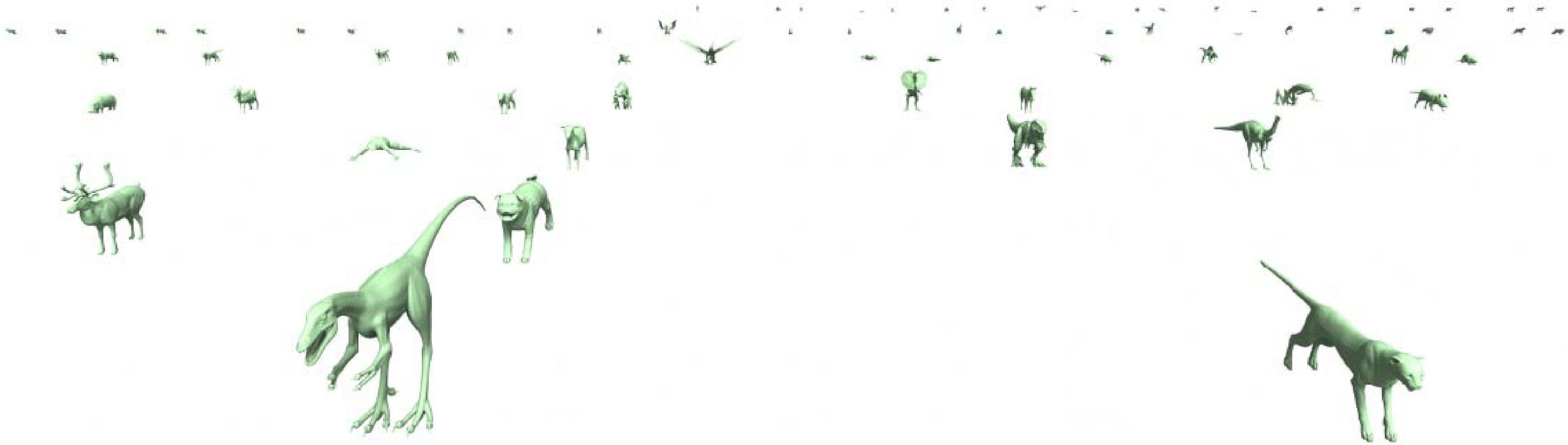


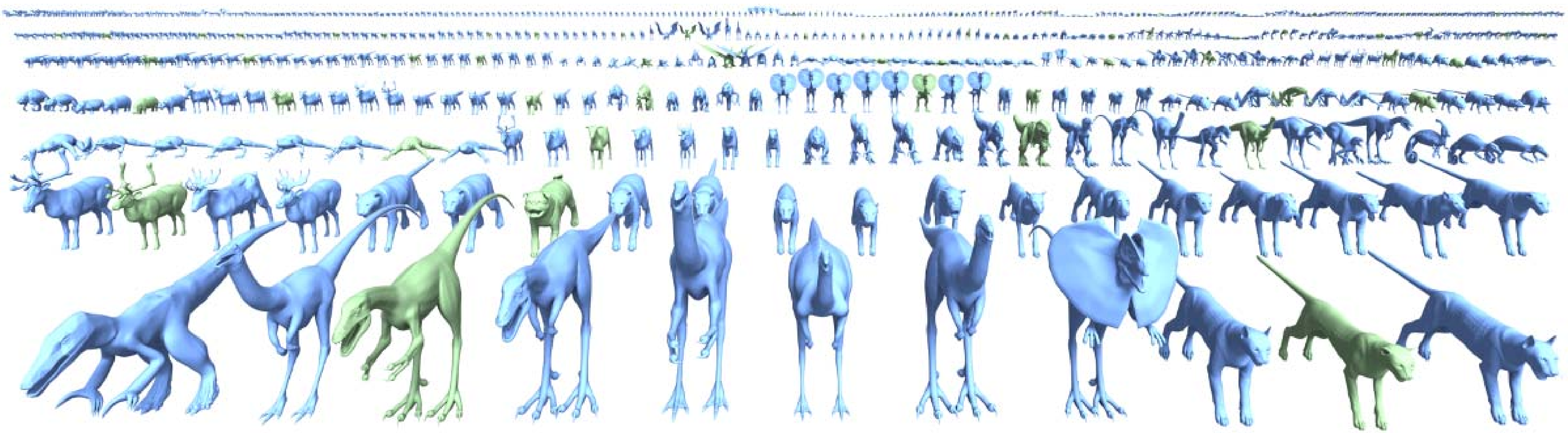


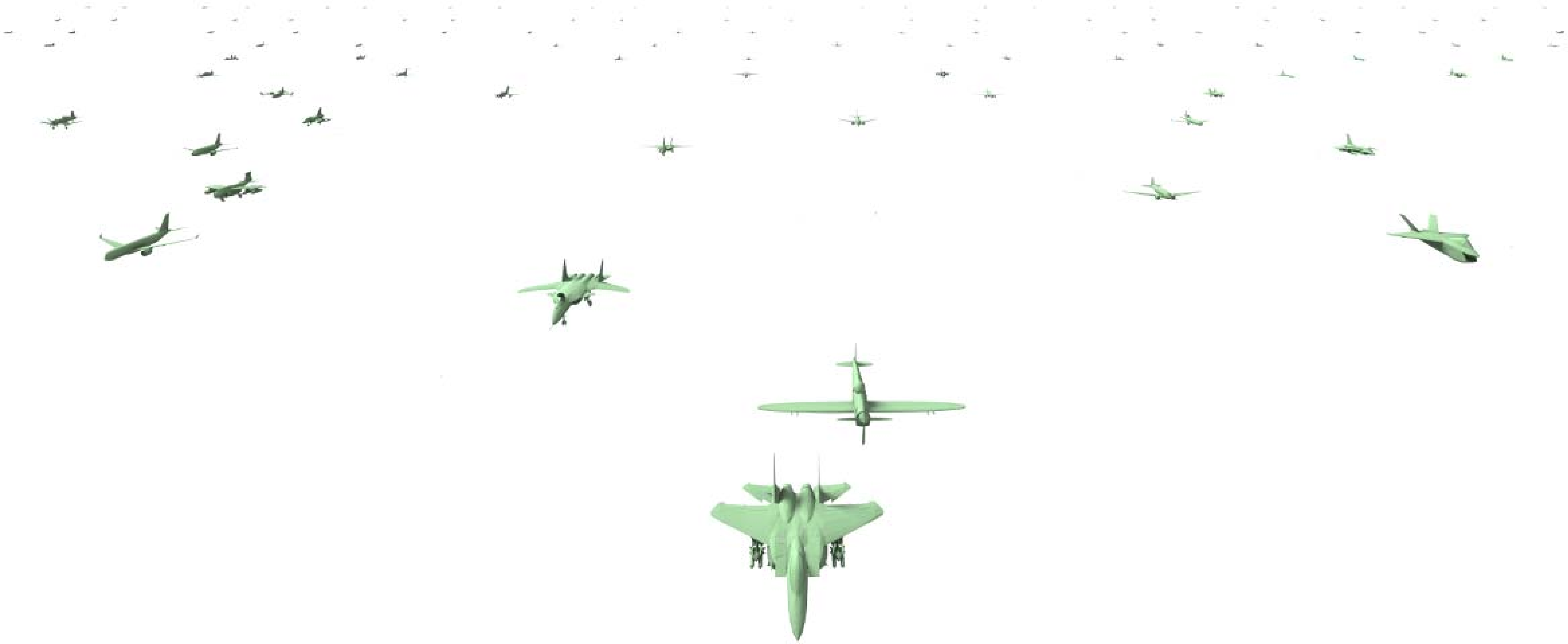


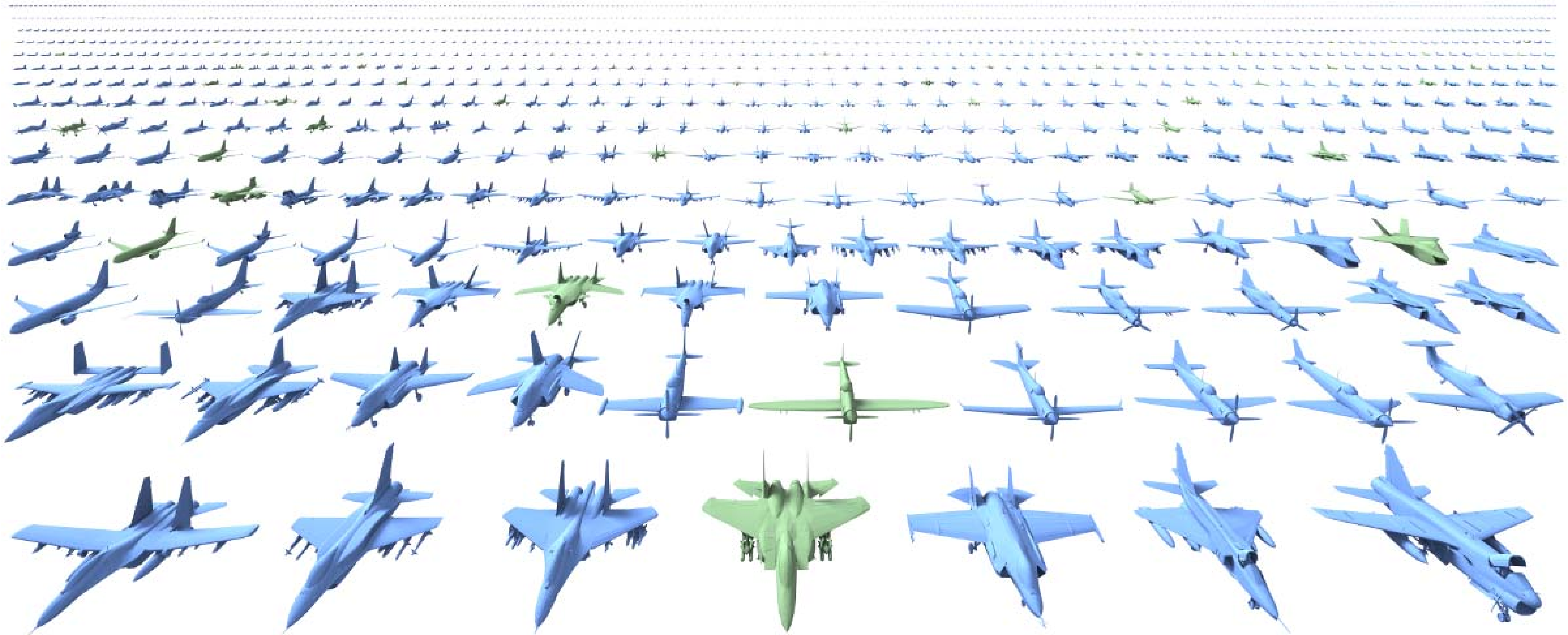




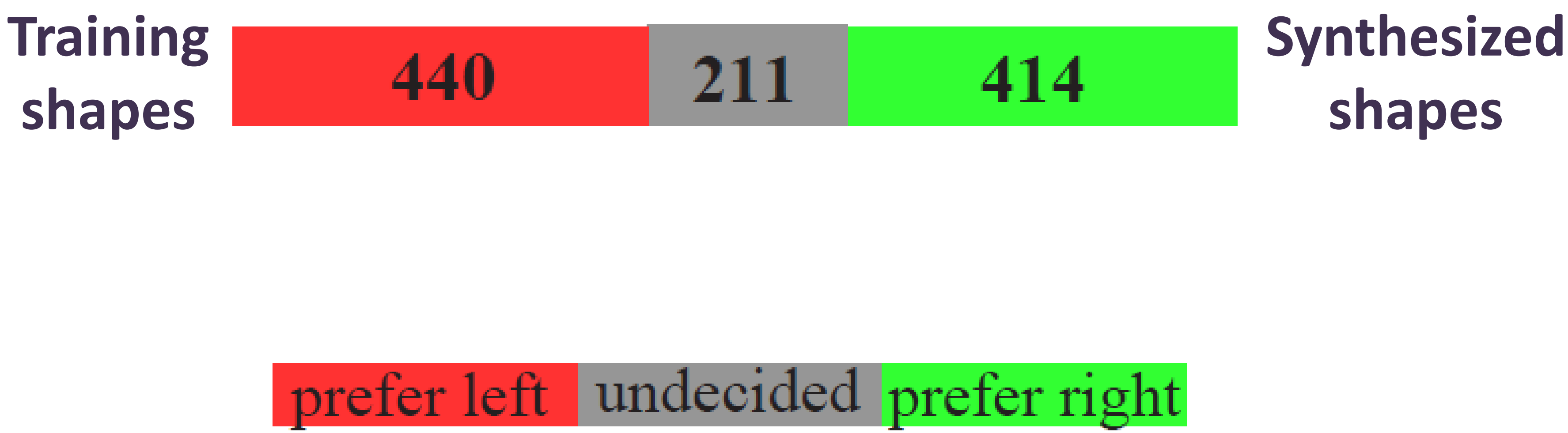








User Survey


Training shapes
440
211
414
Synthesized shapes
prefer left
undecided
prefer right

Constrained shape synthesis

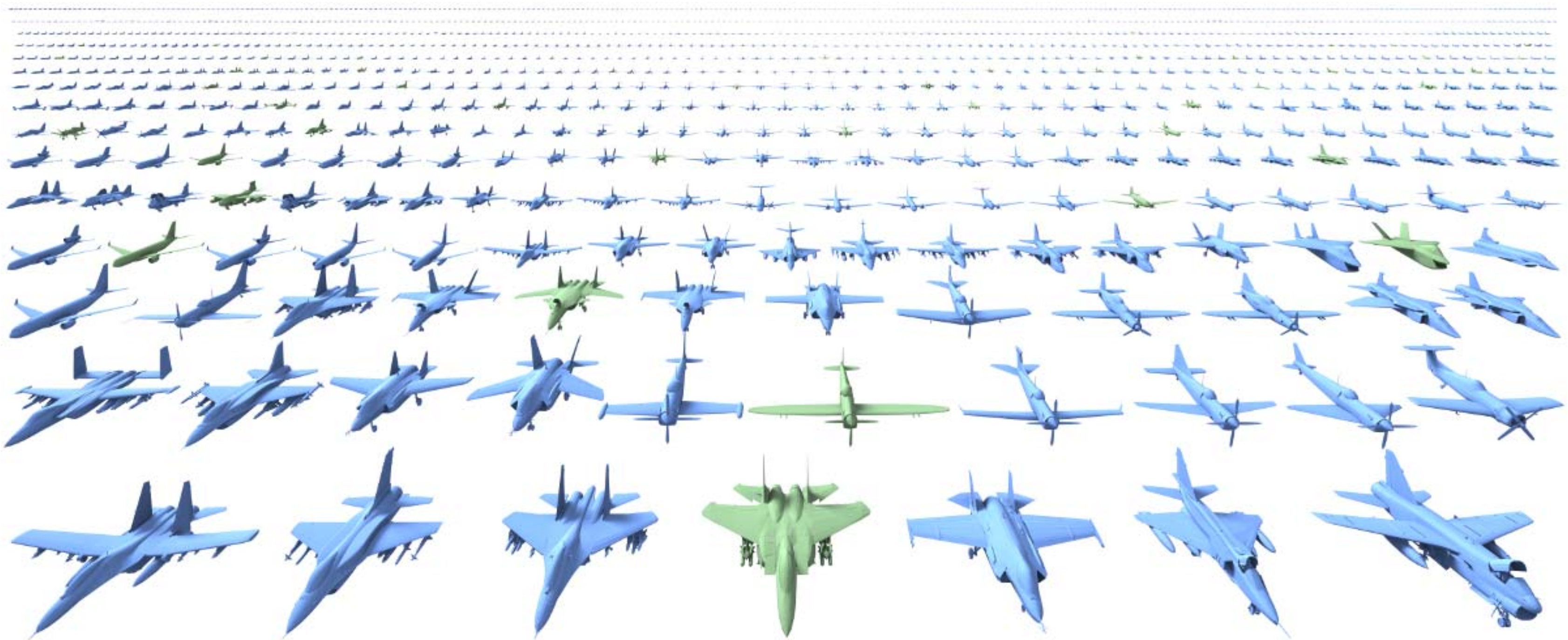
Summary

Generative model of shape structure

Learns structural variability from examples

Applicable to a broad range of complex domains

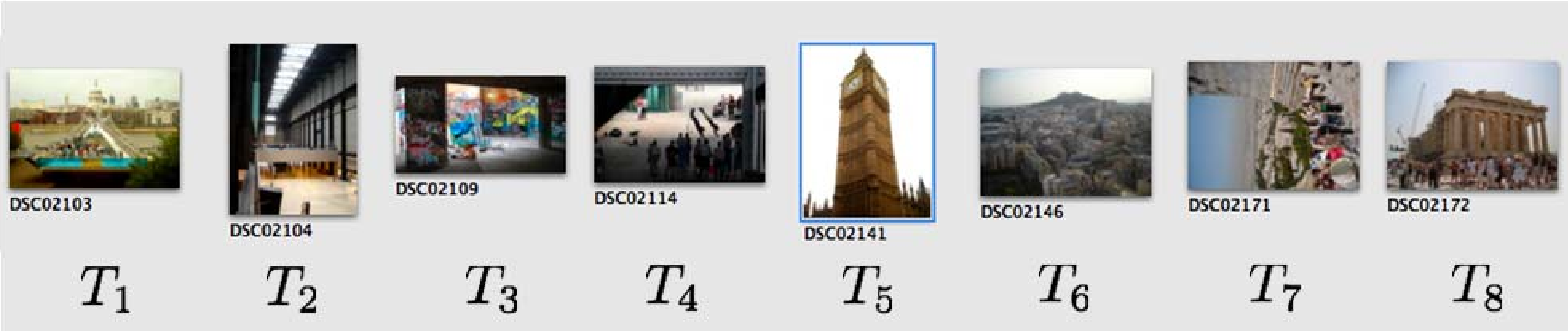
Enables new capabilities for shape processing



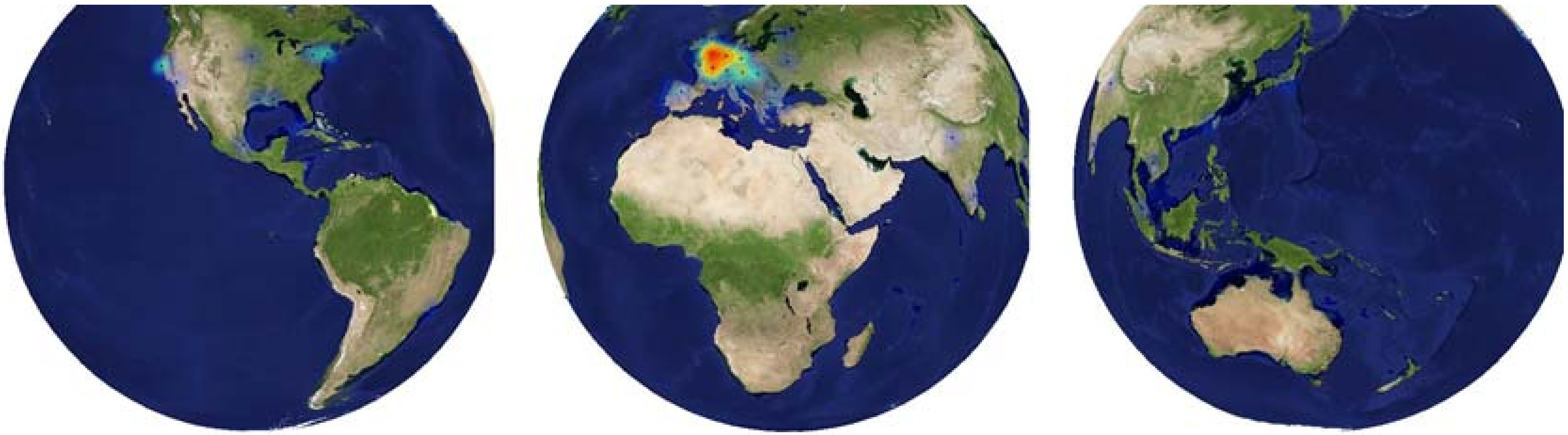
Outline

1. **Learning 3D shape segmentation and labeling**
2. **A generative model of shapes**
3. **Other ML applications to graphics and vision**
4. **Future work**

ML for vision: image sequence geolocation



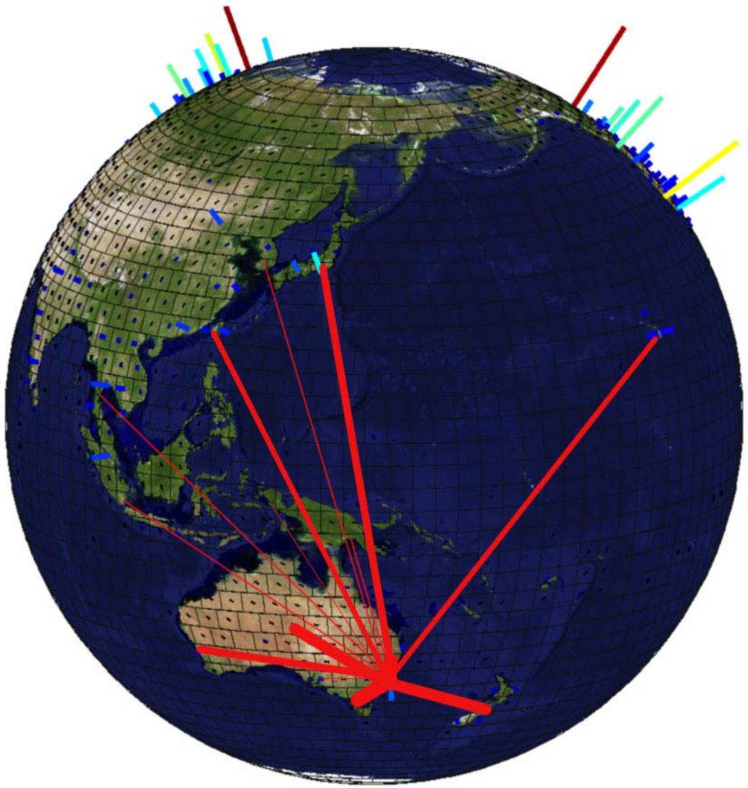
Want: geo-tags



Kalogerakis, Vesselova, Hays, Efros, Hertzmann, ICCV 2009

Image sequence geolocation

How likely are you to travel from one place to another in a given amount of time?



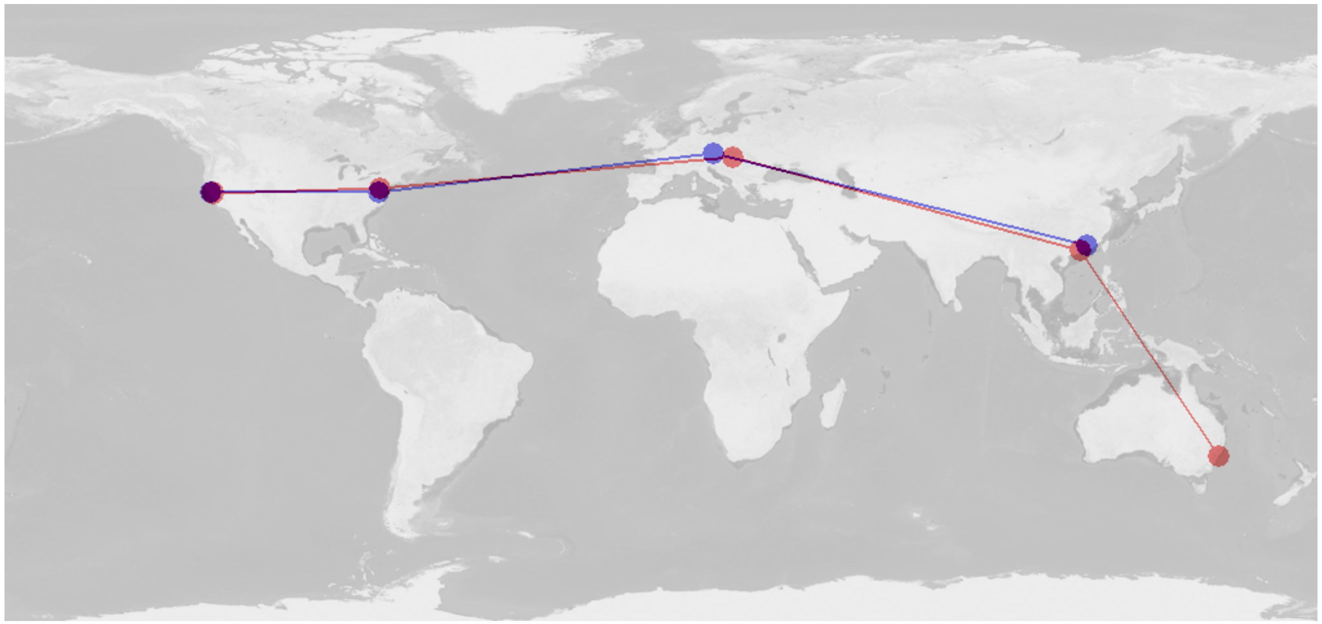
$$P(L_{t+1} = i | L_t = j, \Delta T_t)$$

Image sequence geolocation

137 photos in a user's image sequence

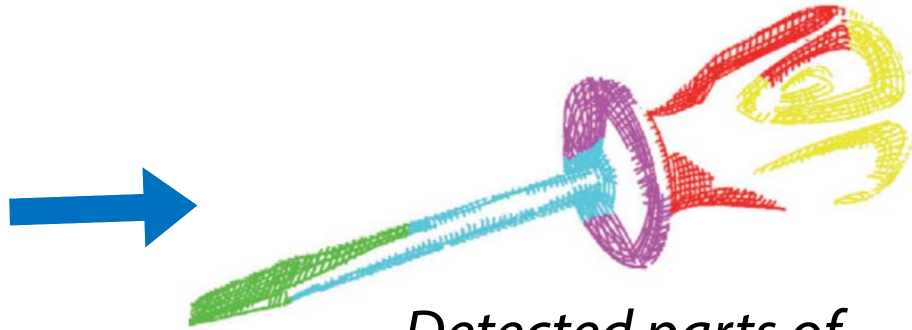
Ground truth path

Estimated path



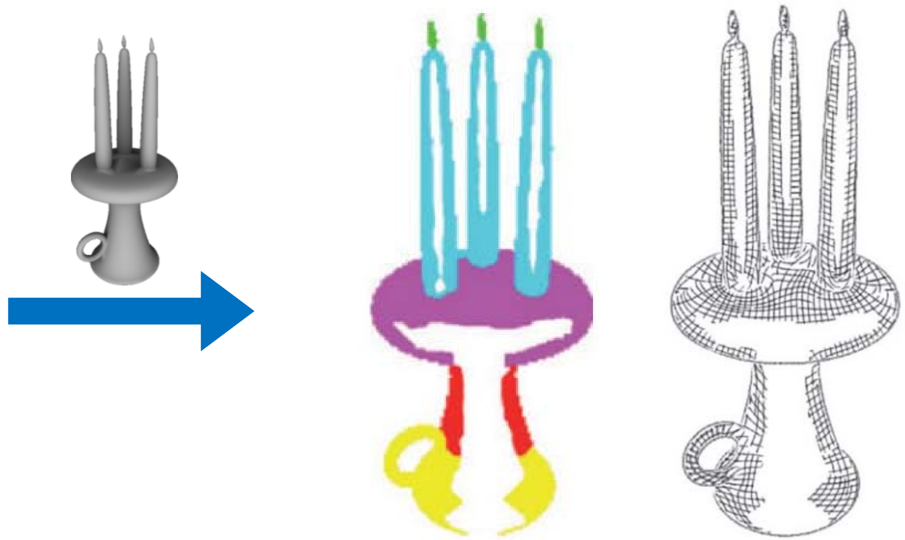
Learning hatching styles

Artist's hatching illustration

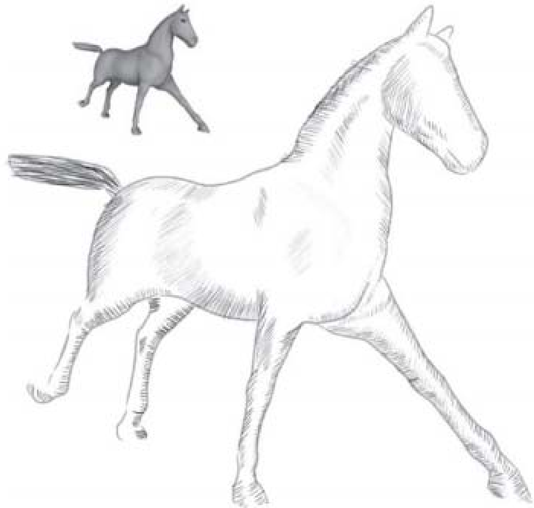


Detected parts of coherent strokes

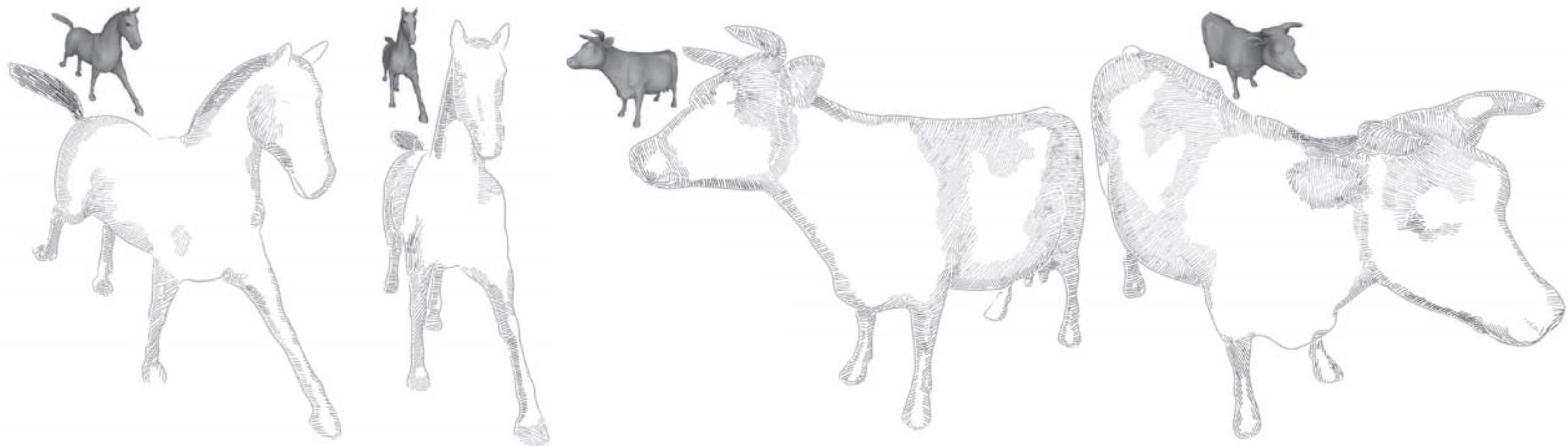
Learned model of stroke properties and parts



Kalogerakis, Nowrouzehahrai, Breslav, Hertzmann
ACM Transactions on Graphics 2012



Training illustration



Generalization to novel views and objects

Outline

1. **Learning 3D shape segmentation and labeling**
2. **A generative model of shapes**
3. **Other ML applications to graphics and vision**
4. **Future work**

Shape understanding in the wild

KinectFusion
[Izadi et al., UIST 2011]

Research goals

Advance shape understanding:

Joint shape recognition and segmentation

Hierarchical shape categorization

Map NL to shapes and deformation handles

Understand function from shapes, print 3D functional shapes

Generative models for:

Variability in symmetries

Architecture

Entire scenes

Images and shapes

Learning algorithms for:

Inferring physical/simulation parameters of shapes

Inferring shape deformations

Texturing, placing lights, other artistic rendering styles

Thank you!

My web page (code, data, demos, videos, papers, etc):

http://people.cs.umass.edu/~kalo/