Learning Probabilistic Models from Collections of 3D Meshes

Sid Chaudhuri, Steve Diverdi, Matthew Fisher, Pat Hanrahan, Vladimir Kim, Wilmot Li, Niloy Mitra, Daniel Ritchie, Manolis Savva, and Thomas Funkhouser
3D Shape Analysis

Database of 3D Meshes → Analysis (Learning) → Probabilistic Model of Shape, Appearance, etc. → Recognition → Reconstruction → Processing → Synthesis → Output 3D Models

Input 3D Meshes → User Input → Output Labels
Computer Vision

Database of Images → Analysis (Learning) → Probabilistic Model of Shape, Appearance, etc. → Recognition → Reconstruction → Processing → Synthesis → Output Labels → Output 3D Models
3D Shape Analysis for Computer Vision?

Database of 3D Meshes → Analysis (Learning) → Probabilistic Model of Shape, Appearance, etc. → Recognition → Reconstruction → Processing → Synthesis → Output Labels → Output 3D Models
Focus of This Talk

This talk will focus on learning probabilistic models of shape from databases of 3D meshes.
Why 3D Shape Analysis?

Why analyze 3D meshes rather than images/scans?

- No noise
- No lighting
- No perspective
- No occlusions
- No pose estimation
- Easier segmentation
- Enough availability
- Large variety

Trimble 3D Warehouse
Why 3D Shape Analysis?

Why analyze 3D meshes rather than images/scans?

- No noise
- No lighting
- No perspective
- No occlusions
- No pose estimation
- Easier segmentation
- Enough availability
- Large variety

Quality?

Trimble 3D Warehouse
Why 3D Shape Analysis?

Why analyze 3D meshes rather than images/scans?

- No noise
- No lighting
- No perspective
- No occlusions
- No pose estimation
- Easier segmentation
- Enough availability
- Large variety

> Quality?
Related Work

Using databases of meshes for scene understanding

- **Fitting 3D meshes to images**
  - Lai 2009, Xu 2011, Satkin 2013, Aubry 2014, etc.
- **Fitting 3D meshes to range scans**
  - Nan 2012, Shen 2012, Kim 2012, Song 2014, etc.
- **Using 3D meshes to learn parameters**
  - Zhao 2013, etc.

Analyzing databases of meshes

- **Consistent segmentation, labeling, correspondence, ...**
  - Golovinskiy 2009, Sidi 2011, Kim 2013, Mitra 2013, etc.
- **Learning probabilistic models**
Outline of Talk

Introduction

Learning probabilistic models from 3D collections
  - Part-based templates
  - Generative model

Conclusions
Outline of Talk

Introduction

Learning probabilistic models from 3D collections
  ➢ Part-based templates
    ◦ Generative model

Conclusions

Goal for This Project

Database of 3D meshes representing an object class

Consistent part segmentations, labels, and correspondences

Probabilistic Model of Shape
Goal for This Project

Database of 3D meshes representing an object class

Challenge

Need to discover segmentations, labels, correspondences, and deformation modes all together

Probabilistic Model of Shape

Consistent part segmentations, labels, and correspondences
Part-Based Templates

Represent object class by part-based templates where each template has a set of parts, and each part has probability distributions for its shape, position, and anisotropic scales.
Template Learning and Fitting

Aim to learn a set of corresponding templates that provides a good fit to every mesh in the database.
Template Learning and Fitting

Aim to learn a set of corresponding templates that provides a good fit to every mesh in the database.
Template Learning and Fitting

Aim to learn a set of corresponding templates that provides a good fit to every mesh in the database.
Template Fitting Problem

For a given template and mesh, aim to minimize:

\[ E = E_{\text{data}} + \gamma E_{\text{deform}} + \beta E_{\text{smooth}} \]

- \( E_{\text{data}} \) (template ↔ shape distance + local shape features)
- \( E_{\text{deform}} \) (plausibility of template deformation)
- \( E_{\text{smooth}} \) (close & similar regions get same label)

Unknowns are:

- Point segmentations and labels
- Point correspondences
- Part center positions
- Part anisotropic scales
Template Fitting Algorithm

Solve by iteratively minimizing different energy terms:

- Segmentation and labeling
- Point correspondence
- Part-aware deformation

\[ E = E_{\text{data}} + \gamma E_{\text{deform}} + \beta E_{\text{smooth}} \]
Template Fitting Algorithm

Solve by iteratively minimizing different energy terms:

- Segmentation and labeling
  - Point correspondence
  - Part-aware deformation

\[ E = E_{\text{data}} + \gamma E_{\text{deform}} + \beta E_{\text{smooth}} \]

Solve with graph cut [Boykov 2001]
Template Fitting Algorithm

Solve by iteratively minimizing different energy terms:

- Segmentation and labeling
  - Point correspondence
- Part-aware deformation

\[ E = E_{\text{data}} + \gamma E_{\text{deform}} + \beta E_{\text{smooth}} \]

Solve with part-aware closest points
Template Fitting Algorithm

Solve by iteratively minimizing different energy terms:

- Segmentation and labeling
- Point correspondence
- Part-aware deformation

\[ E = E_{\text{data}} + \gamma E_{\text{deform}} + \beta E_{\text{smooth}} \]

Solve for positions and scales of each part by setting partial derivatives to zero.
Template Learning Problem

Aim to **learn** a set of corresponding templates that provides a good fit to every mesh in the database.
Template Learning Algorithm

Iteratively grow a set of templates with each optimized to fit a different cluster of meshes

Template Initialization

Template Fitting

Template Refinement

repeat until convergence
Template Learning Algorithm

- Shapes
- Fitting Set
- Templates

Template Initialization
Template Fitting
Template Learning Algorithm

Template Initialization

Template Fitting
Template Learning Algorithm

Template Initialization

Template Fitting

Learning Set
Template Learning Algorithm

Template Initialization

Template Fitting

Template Refinement
Template Learning Algorithm

Template Initialization

Template Fitting

Template Refinement
Template Learning Algorithm

Template Initialization

Template Fitting

Template Refinement

repeat until convergence
Template Learning Example

Initial Part-Based Template
Template Learning Example

Updated Part-Based Template
Template Learning Example

New Part-Based Template
Template Learning Example
Template Learning and Fitting Results

Data sets:
- Crawl SketchUp Warehouse for collections by keyword
- Eliminate outliers with Mechanical Turk
- Specify manual correspondences for subset of models

Experiments:
- Solve for part-based templates for collection
- Evaluate correspondences & segmentations
Template Learning and Fitting Results

2 Templates

1508

1605

3113 Airplanes
Template Learning and Fitting Results

2 Templates

1508

1605

3113 Airplanes

Failure
Template Learning and Fitting Results

2 Templates

441 Bikes
Template Learning Results

2 Templates

378

63

441 Bikes

Failures
Surface Correspondence Results

Correspondence benchmark (7442 seats)
# Surface Segmentation Results

Co-segmentation benchmark [Sidi et al, 2011]

<table>
<thead>
<tr>
<th>Class</th>
<th>Hu</th>
<th>Our</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chairs</td>
<td>89.6</td>
<td>97.6</td>
</tr>
<tr>
<td>Lamps</td>
<td>90.7</td>
<td>95.2</td>
</tr>
<tr>
<td>FourLegged</td>
<td>88.7</td>
<td>86.9</td>
</tr>
<tr>
<td>Goblets</td>
<td>99.2</td>
<td>97.6</td>
</tr>
<tr>
<td>Vase</td>
<td>80.2</td>
<td>81.3</td>
</tr>
<tr>
<td>Guitars</td>
<td>98.0</td>
<td>88.5</td>
</tr>
<tr>
<td>Candelabra</td>
<td>93.9</td>
<td>82.4</td>
</tr>
</tbody>
</table>

Within 2% or ours is better
Outline of Talk

Introduction

Learning probabilistic models from 3D collections
  - Part-based templates
  - Generative model

Conclusions
Goal for This Project

Exemplar scenes + Database of Scenes → Probabilistic Model of Shape → Synthesized novel scenes
Goal for This Project

Exemplar scenes + Database of Scenes → Probabilistic Model of Shape

Challenge
Need to learn a model with great generality from few examples

Synthesized novel scenes
Define categories of objects based on their contexts in a scene rather than basic functions

- Learned from examples by clustering of objects with similar spatial neighborhoods
Generative Model

Represent the probability of a scene $S$ by a generative model based on category cardinalities ($c$), support hierarchy topology relationships ($t$), and spatial arrangement relationships ($a$)

$$P(S) = P(c, t, a) = P(a|t, c) P(t|c) P(c)$$

Exemplar scenes
Generative Model Details

Category cardinalities: $P(c)$
- Represent with Bayesian network
- Boolean random variables (# desks > 1?)
- Add support surface constraints

Object frequencies in target scenes + support constraints

Bayesian network
Support relationships: $P(t|c)$

- Boolean random variables (desk supports keyboard?)
- Learn frequencies for pairs of categories
- Total probability is product over all objects in scene

$$P(t|c) = \prod_o P(C(o), C(\text{support}(o)))$$
Generative Model Details

Spatial arrangements: \( P(a|t,c) = R(a,t,c)S(a,t,c) \)

- Random variables for relative positions and orientations
- Pairwise distributions of spatial relationships

Distributions of spatial relationships for pairs of object categories
Generative Model Details

Spatial arrangements: $P(a|t,c) = R(a,t,c)S(a,t,c)$
- Random variables for relative positions and orientations
- Pairwise distributions of spatial relationships
- Feature distributions for positions on support surfaces

Distributions of geometric features of support surfaces
Scene Synthesis

Exemplar scenes + Database of Scenes → Probabilistic Model of Shape → Synthesized novel scenes
Scene Synthesis Results

Synthesized novel scenes
Scene Synthesis Results

User study suggests that people find our synthesized scenes almost as good as manually created ones.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Manual</strong> Dining Tables</td>
<td>3.83</td>
<td>3.50</td>
<td>3.51</td>
<td>4.03</td>
<td>4.27</td>
</tr>
<tr>
<td><strong>Synth</strong> Dining Tables</td>
<td>3.98</td>
<td>4.03</td>
<td>4.03</td>
<td>4.03</td>
<td>4.03</td>
</tr>
<tr>
<td><strong>Manual</strong> Gamer Desks</td>
<td>3.50</td>
<td>3.50</td>
<td>3.51</td>
<td>4.03</td>
<td>4.27</td>
</tr>
<tr>
<td><strong>Synth</strong> Gamer Desks</td>
<td>3.98</td>
<td>4.03</td>
<td>4.03</td>
<td>4.03</td>
<td>4.03</td>
</tr>
<tr>
<td><strong>Manual</strong> Office Desks</td>
<td>3.83</td>
<td>3.50</td>
<td>3.51</td>
<td>4.03</td>
<td>4.27</td>
</tr>
<tr>
<td><strong>Synth</strong> Office Desks</td>
<td>3.98</td>
<td>4.03</td>
<td>4.03</td>
<td>4.03</td>
<td>4.03</td>
</tr>
</tbody>
</table>
Outline of Talk

Introduction

Learning probabilistic models from 3D collections
- Part-based templates
- Generative model

Conclusions
Conclusions

Main result:
- Probabilistic models can be learned from collections of 3D meshes

Future work:
- It will be interesting to see if these models can be used effectively to understand scans and images
3D Shape Analysis for Computer Vision?

Database of 3D Meshes → Analysis (Learning) → Probabilistic Model of Shape, Appearance, etc. → Recognition

Input Images → Output Labels

User Input → Output 3D Models

→ Reconstruction

→ Processing

→ Synthesis
Acknowledgments

People:
  ◦ Sid Chaudhuri, Steve Diverdi, Matthew Fisher, Pat Hanrahan, Vladimir Kim, Wilmot Li, Niloy Mitra, Daniel Ritchie, Manolis Savva

Funding:
  ◦ NSF, NSERC, Intel, Adobe, Google

Data sets:
  ◦ Trimble 3D Warehouse

Thank You!