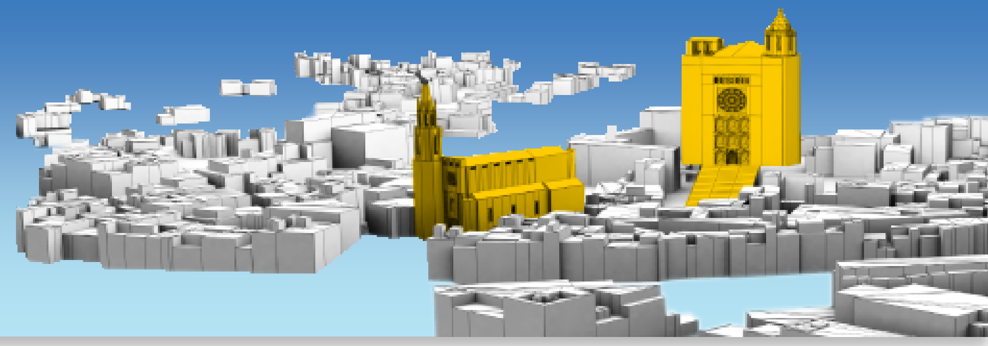
Eurographics 2013
May 6-10, Girona (Spain)



Structure-Aware Shape Processing

Part II
Structure Models & Analysis

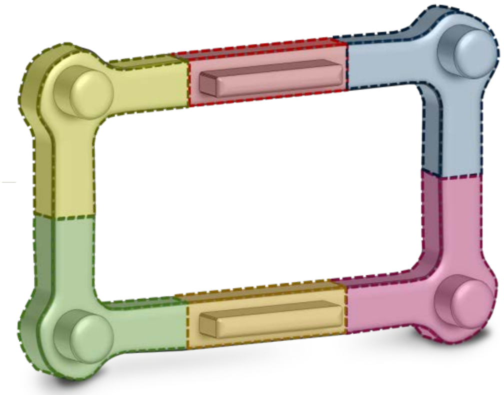
Michael Wand

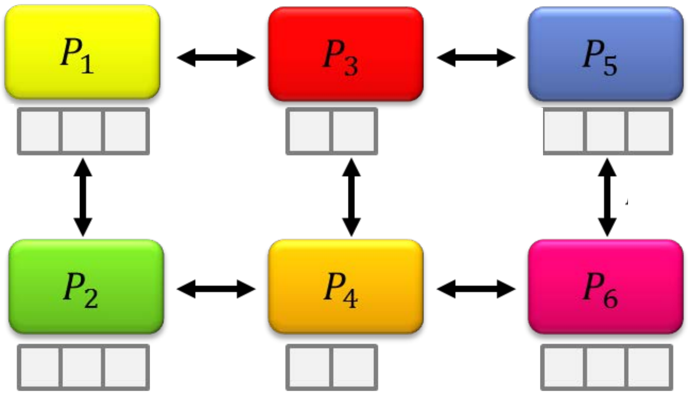
Niloy Mitra *Michael Wand* *Hao Zhang* *Daniel Cohen-Or* *Martin Bokeloh*

Overview

Types of Structure

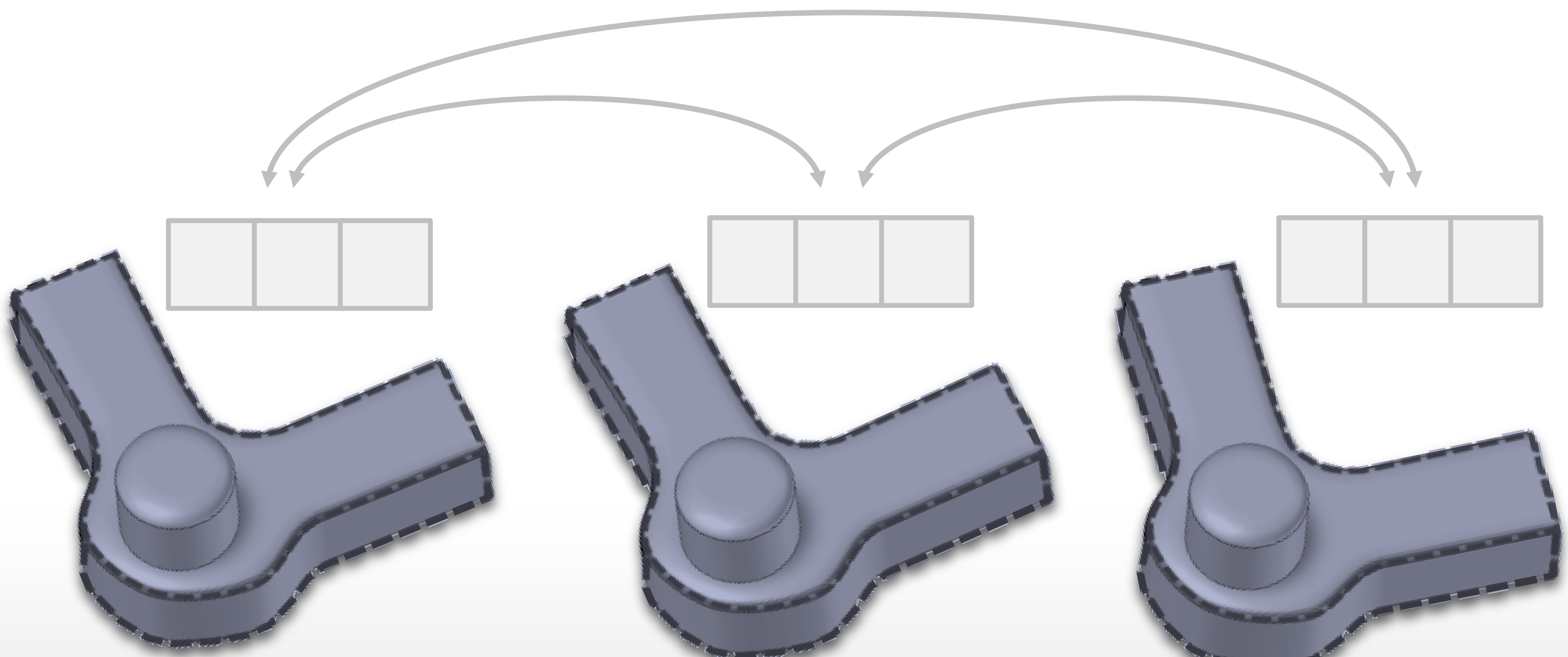
- Model components
  - Parts
  - Parameters
  - Relations
- Classification
  - User input
  - Fixed models
  - Data-driven models





Structure Analysis / Algorithms

1. Parts
2. Parameters
3. Relations



1. Parts

Modeling and Detecting Parts

1.1 User defined parts

- Manual segmentation

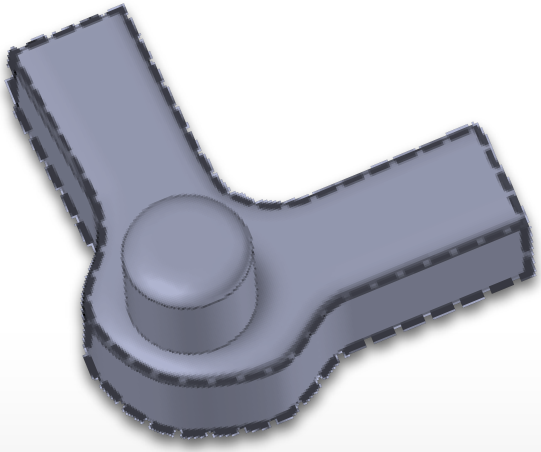
1.2 Fixed models

- A priori segmentation model

1.3 Data-driven segmentation

- A priori: meta-model

Parts



1. Parts

Modeling and Detecting Parts

1.1 User defined parts

- Manual segmentation

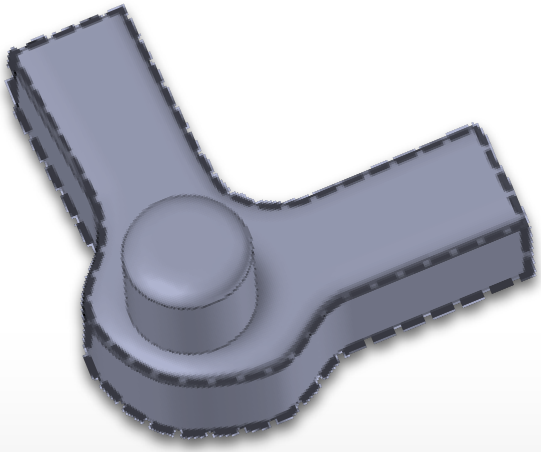
1.2 Fixed models

- A priori segmentation model

1.3 Data-driven segmentation

- A priori: meta-model

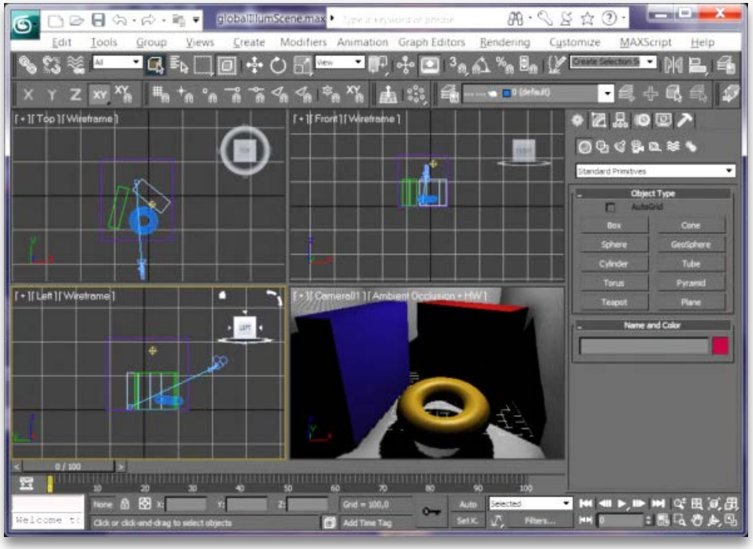
Parts



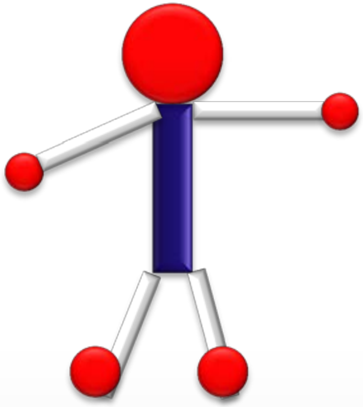
1.1 User Defined Parts

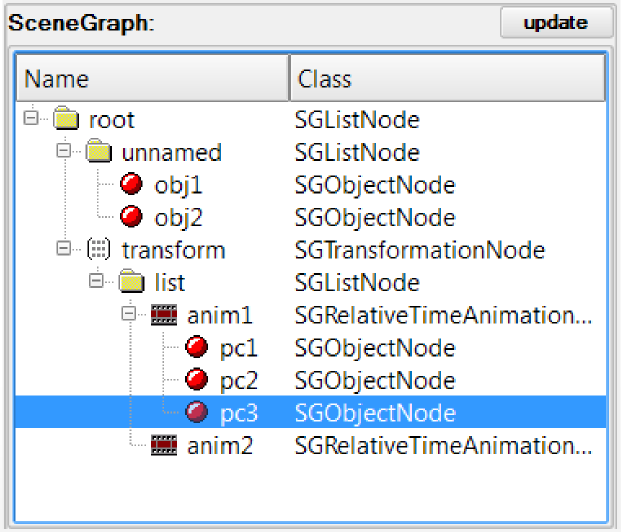
Traditional Modeling

- Scenes assembled out of primitives
- Hierarchical organization in scene graph



[Autodesk 3DS MAX]



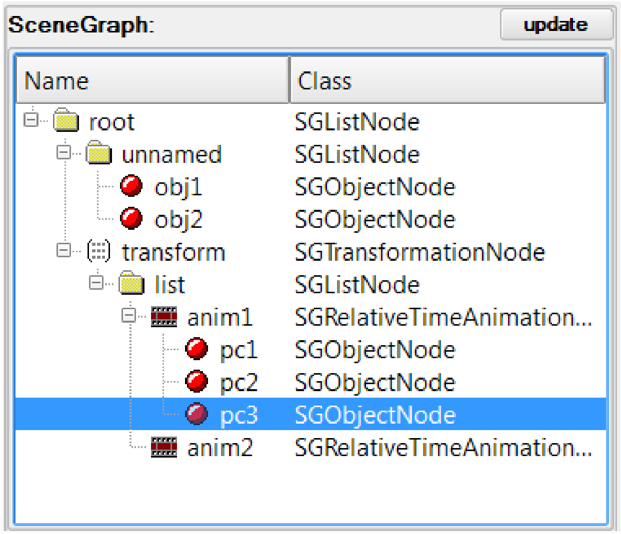


1.1 User Defined Parts

[Jain et al. EG 2012]

Reusing Existing Scene Graphs

- Meshes from model collections
- Artist-created
- Many techniques use provided segmentation / scene graph
- Possibly additional annotation (node names, meta-data)



1. Parts

Modeling and Detecting Parts

1.1 User defined parts

- Manual segmentation

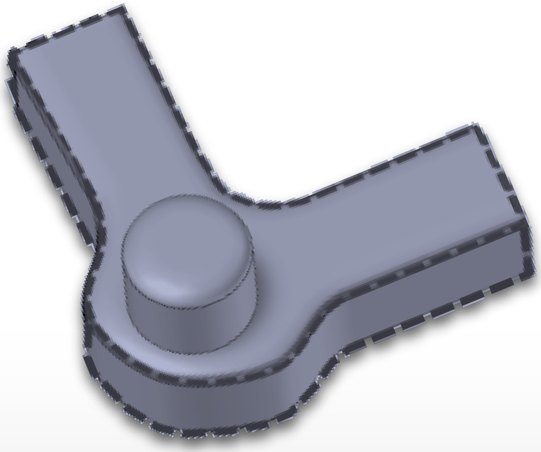
1.2 Fixed models

- A priori segmentation model

1.3 Data-driven segmentation

- A priori: meta-model

Parts

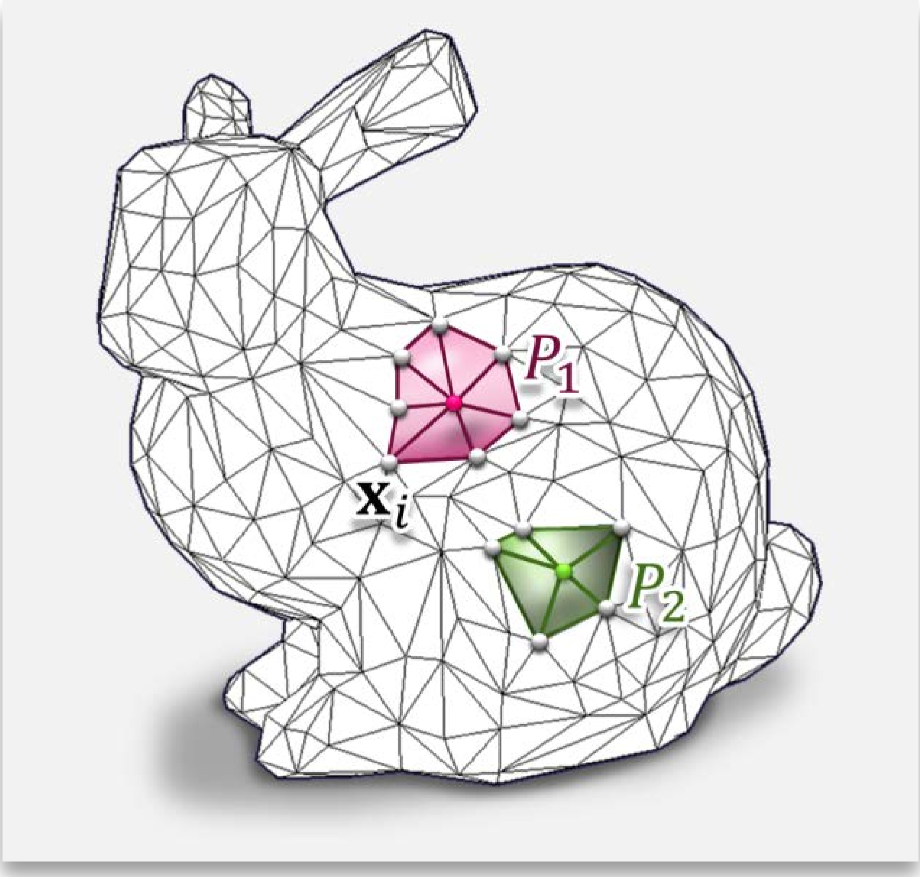


1.2 Parts – Fixed Models

Fixed Models

- Generic discretization
- Feature detection
- Shape segmentation (local)
- Symmetry-based segmentation (global)

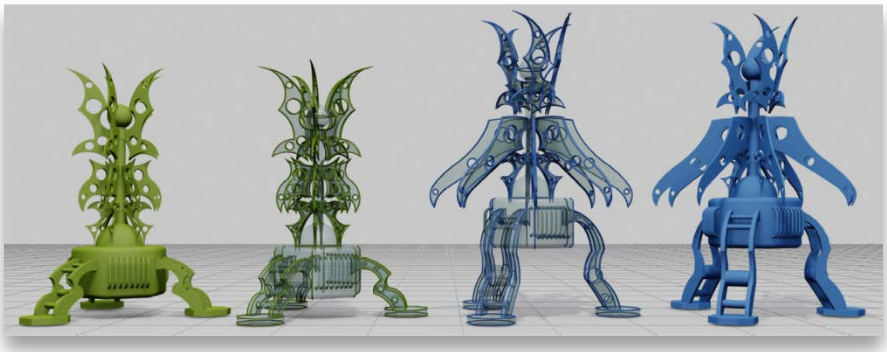
Generic Discretization



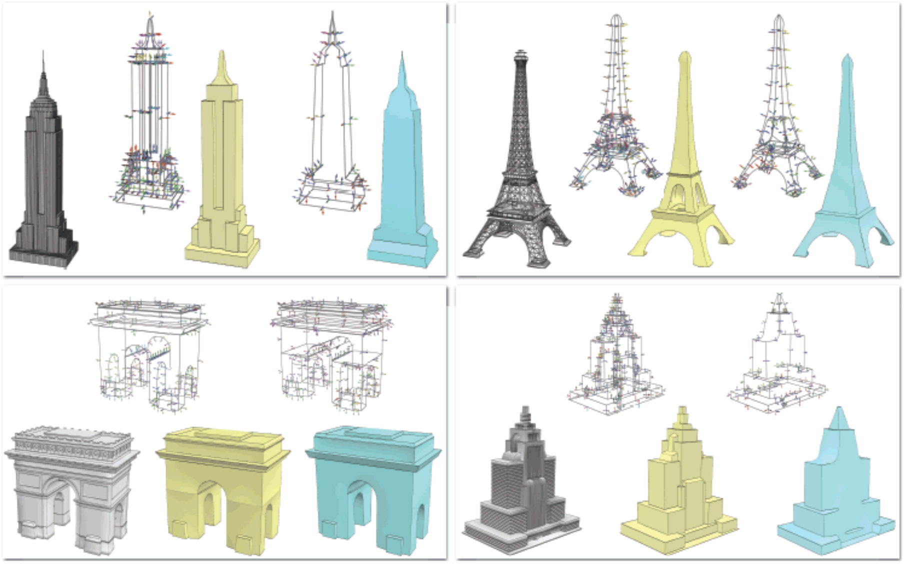
Generic Representations

- E.g. finite elements

Feature detection



[Gal et al. SG 2009]

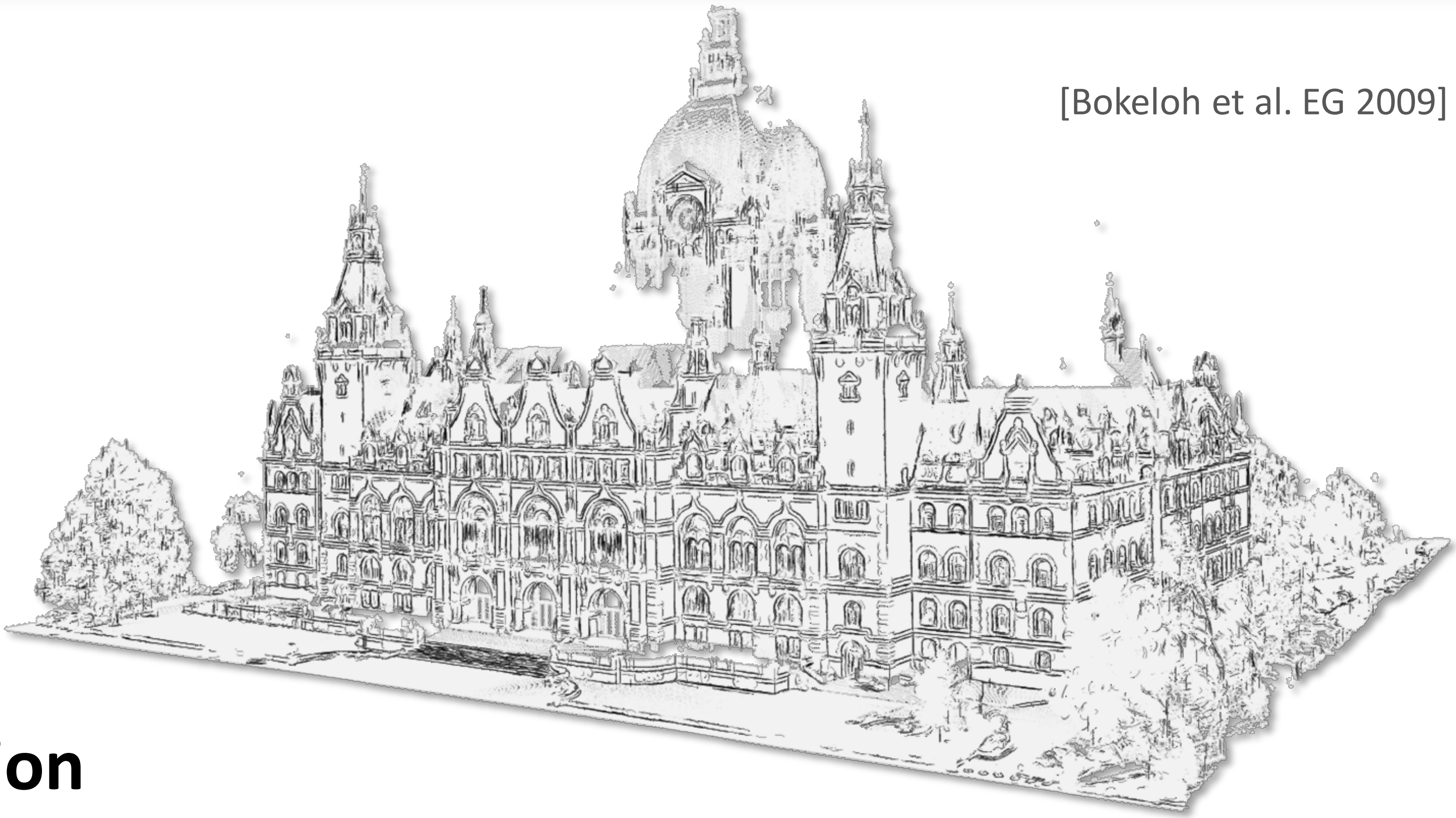


[Mehra et al. 2009]

Features as Parts

- Sharp creases as parts: for example iWires
- Shape abstraction

Feature detection



[Bokeloh et al. EG 2009]

Detection

- Local maxima of principal curvature
- Slippage analysis
- Crease edges (clean meshes), local fitting (scanner data)

Shape Segmentation

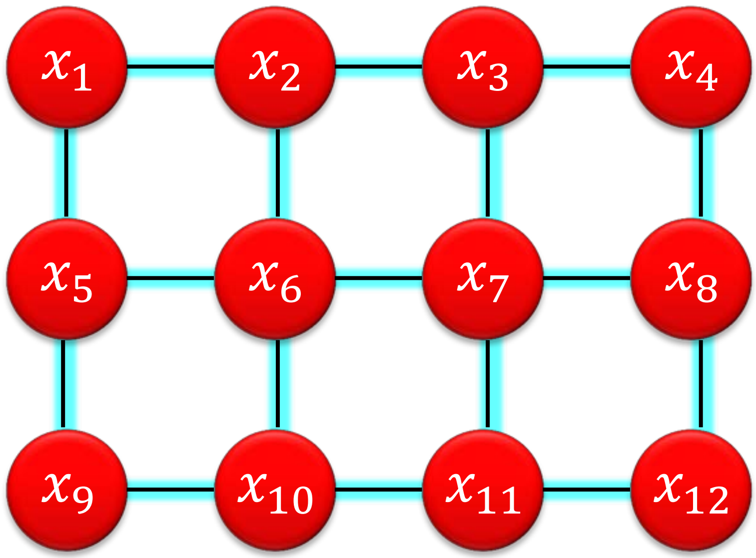
Shape Segmentation

- Well-established field
- Plenty of models

General Strategy

- Local evidence for part labels
  - For each primitive
- Coherent neighborhoods
  - Consider pairs of adjacent primitives
- Markov random field model
  - Optimize assignment

Markovian Graphical Model



$\boxed{p_i^{(1)}(x_i)}$

Different labels for each triangle, pixel, etc.

$\boxed{p_{i,j}^{(2)}(x_i, x_j)}$

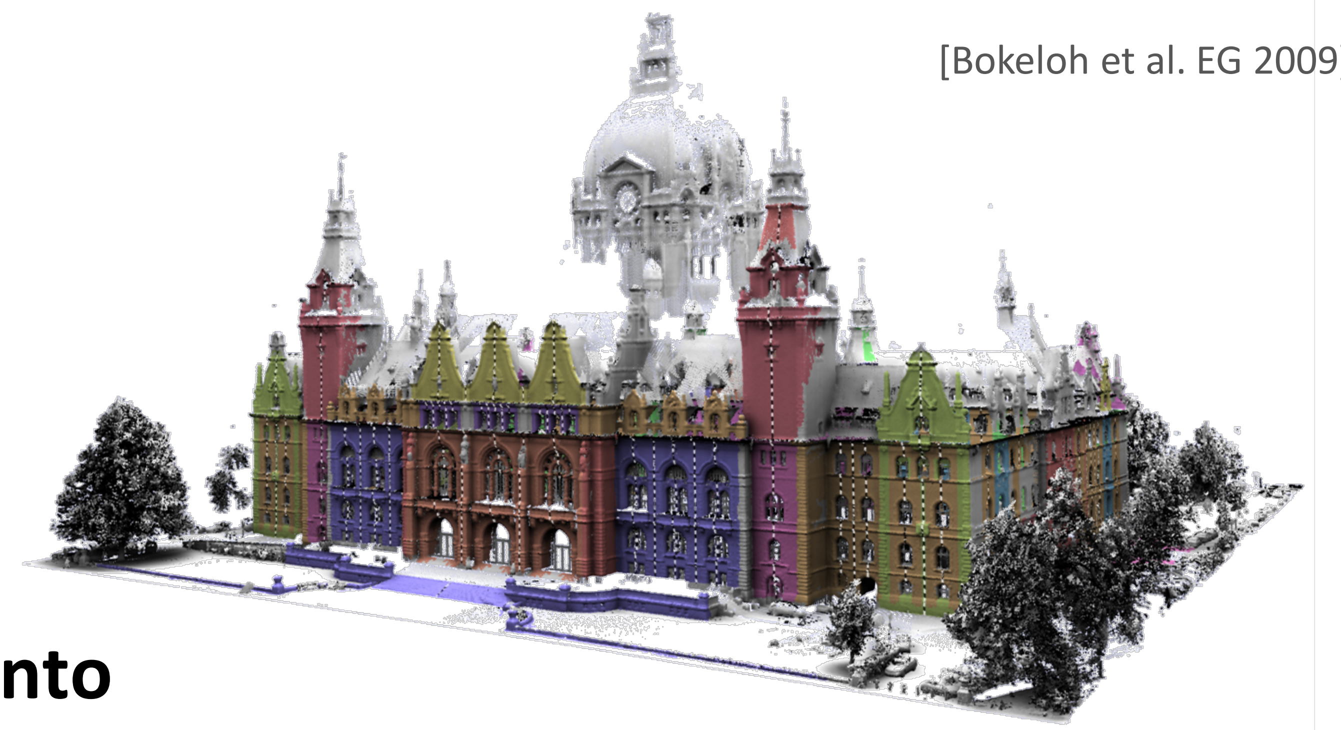
pairs of triangle, pixel, etc.

Pairwise Markov Random Field

$$p(x_1, \ldots, x_n) = \frac{1}{Z} \prod_{i=1}^{n} \underline{p_i^{(1)}(x_i)} \prod_{i,j \in E} \underline{p_{i,j}^{(2)}(x_i, x_j)}$$

- Optimization: loopy belief propagation, (iterated) graph cuts

Symmetry-based Segmentation

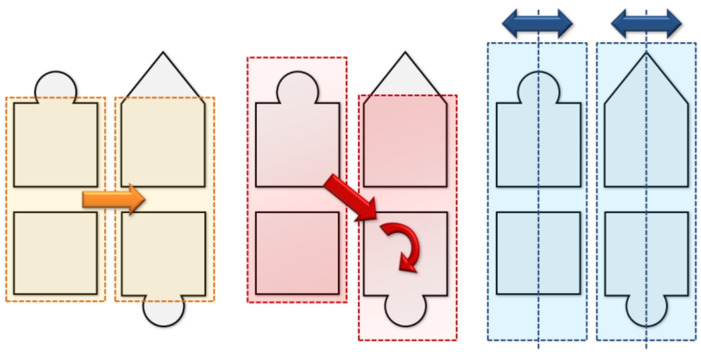


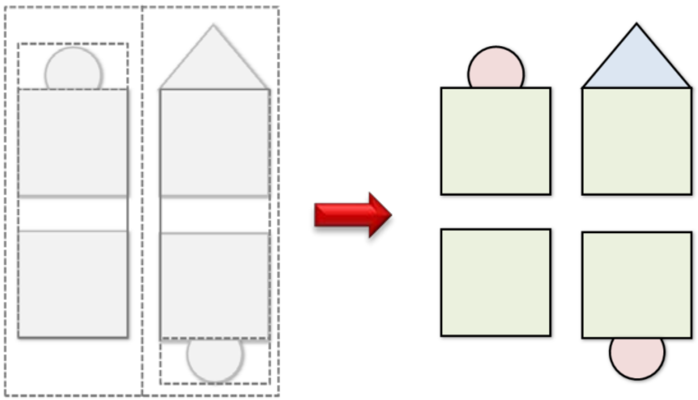
Split Model into Symmetric Building Blocks

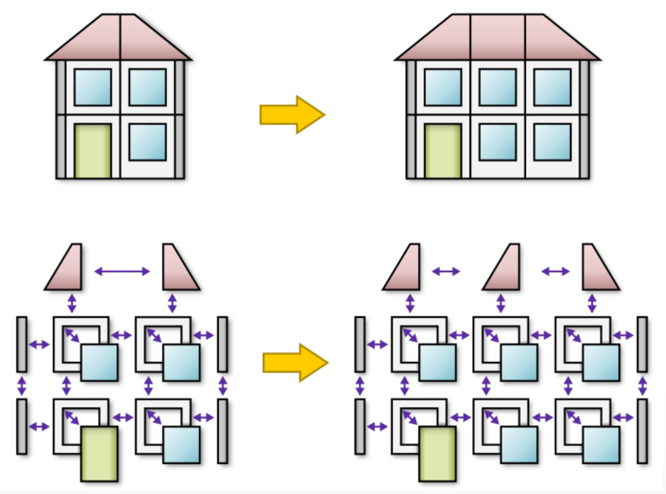
- Group repetitive elements
- Discover redundant instances

⇒ Tutorial last Tuesday

Microtiles

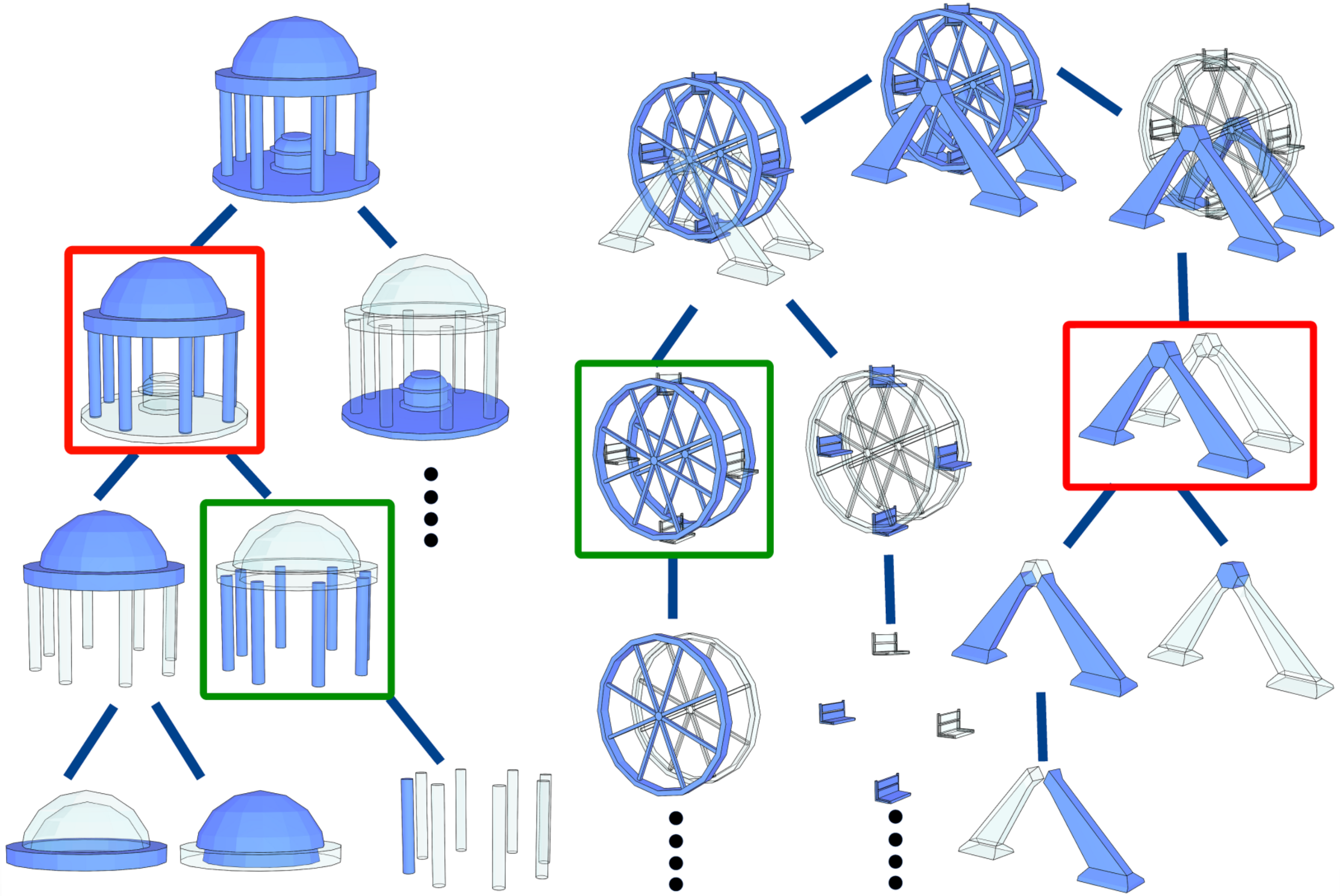






[Kalojanov et al. SGP 2012]

Symmetry Hierarchies



[Wang et al. EG 2011]

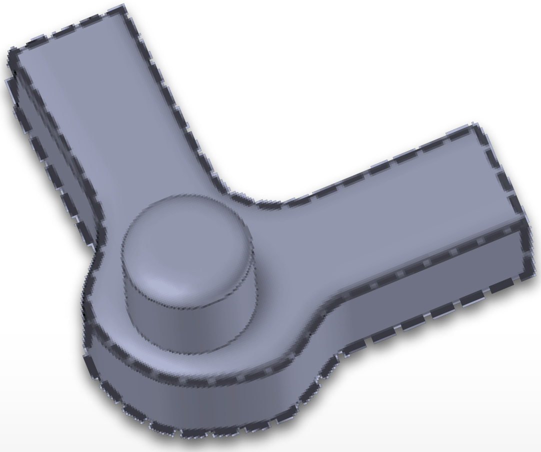
1. Parts

Modeling and Detecting Parts

1.1 User defined parts
- Manual segmentation

1.2 Fixed models
- A priori segmentation model

1.3 Data-driven segmentation
- A priori: meta-model

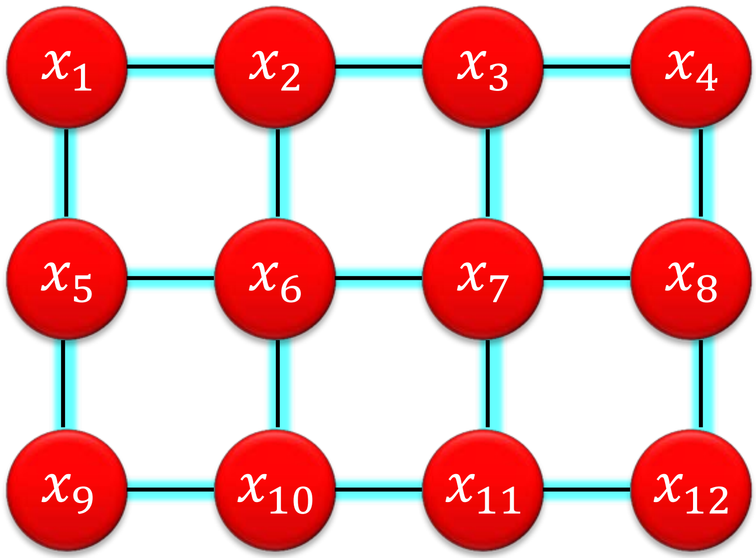


1.3 Learning Parts

Approach

- Supervised
  - Training data: Example Segmentations
  - Learn model parameters
  - Apply to more, unknown data after training
- Unsupervised
  - No training phase
  - Clustering: Maximize coherence
  - Co-segmentation

Supervised Learning



$\boxed{p_i^{(1)}(x_i)}$

Learn: Local evidence

$\boxed{p_{i,j}^{(2)}(x_i, x_j)}$

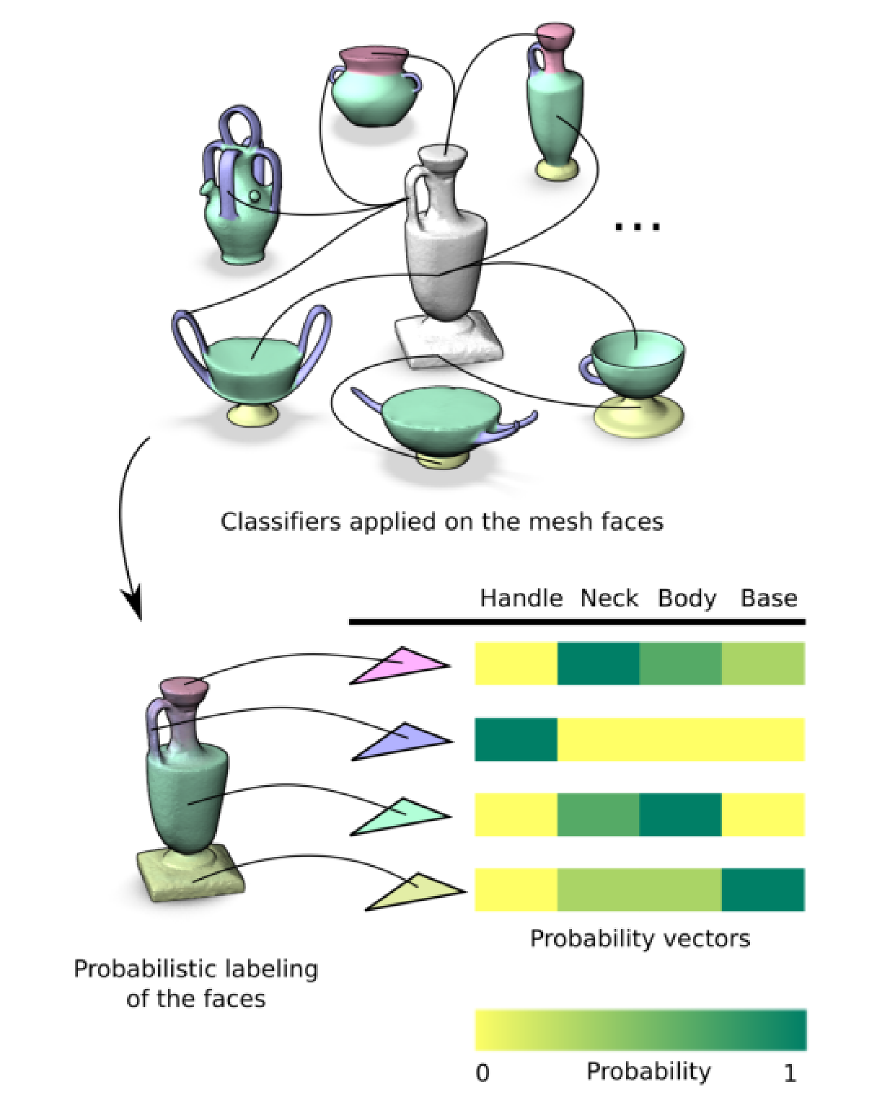
Learn: pairwise relations

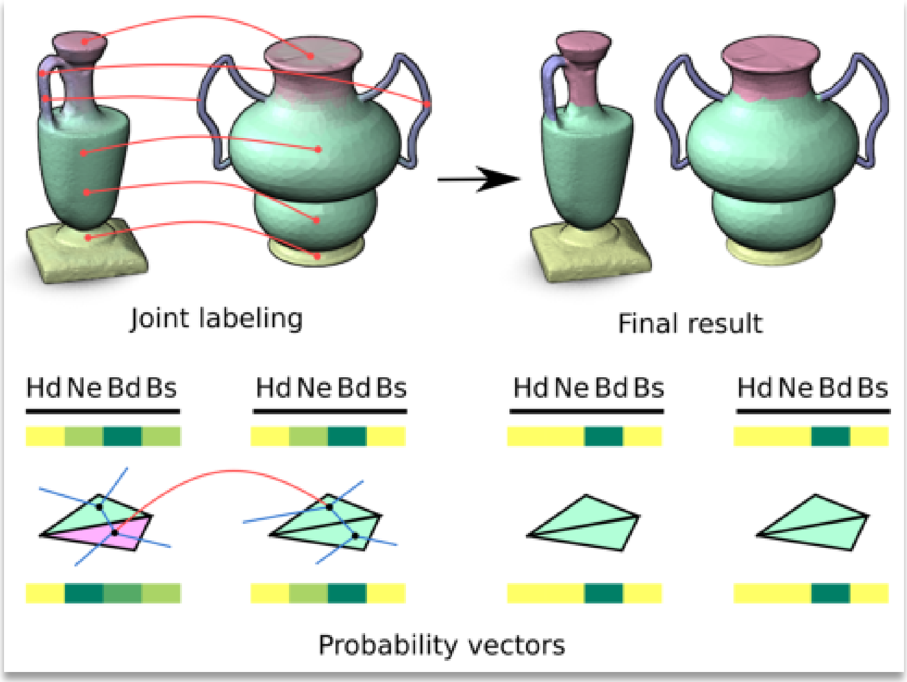
Markov (Conditional) Random Field

$$p(x_1, \dots, x_n) = \frac{1}{Z_\theta} \prod_{i=1}^{n} p_i^{(1)}(x_i|\theta) \prod_{i,j \in E} p_{i,j}^{(2)}(x_i, x_j|\theta)$$

- Optimize same function, but for parameters θ.

Unary and Pairwise Potentials





$p^{(2)}$ – **pairwise**

$p^{(1)}$ – **unary**

[van Kaik et al. EG 2011]

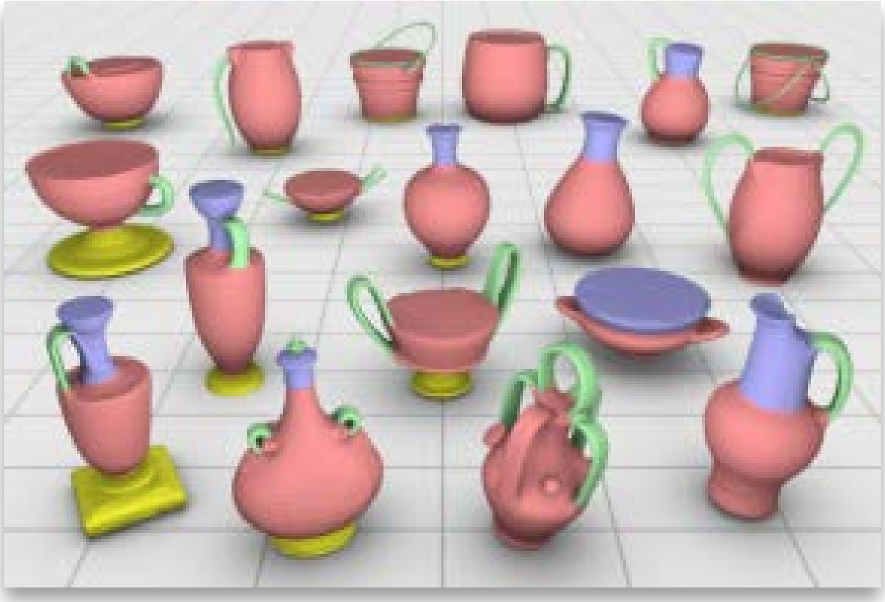
Results

[van Kaik et al. EG 2011]

Approaches

- [Kalogerakis et al. 2010, van Kaik et al. 2011, Laga et al. 2013]
- Approximation: Learn $p^{(1)}, p^{(2)}$ separately
- [Anguelov et al. 2005]: full optimization for special model

Unsupervised Learning



[Sidi et al. SIGA 2011]

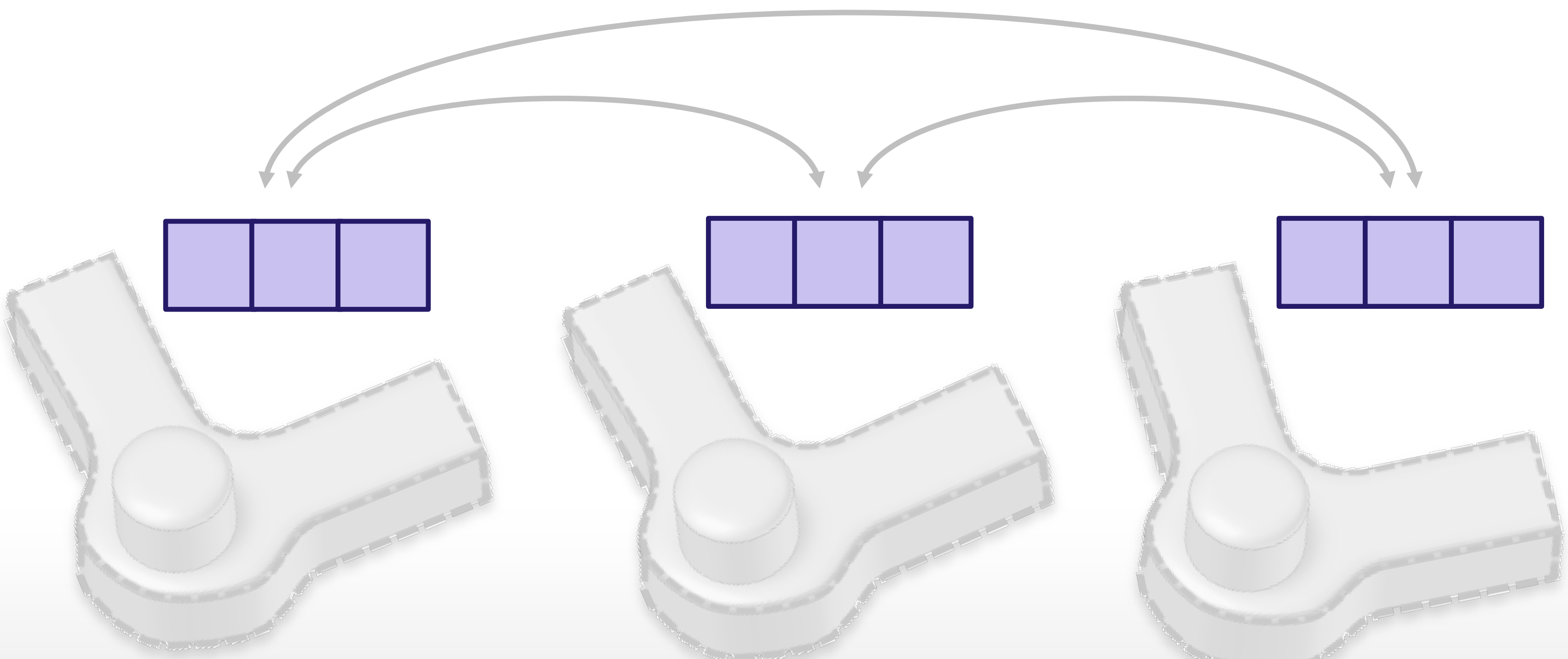
Approaches

- Similar Markov-random field model
- Data-term: Clustering in descriptor space
- Pairwise term: a priori model

1. Parts

2. Parameters

3. Relations



2. Parameters

Modeling and Detecting Parameters of Parts

2.1 User defined parameters

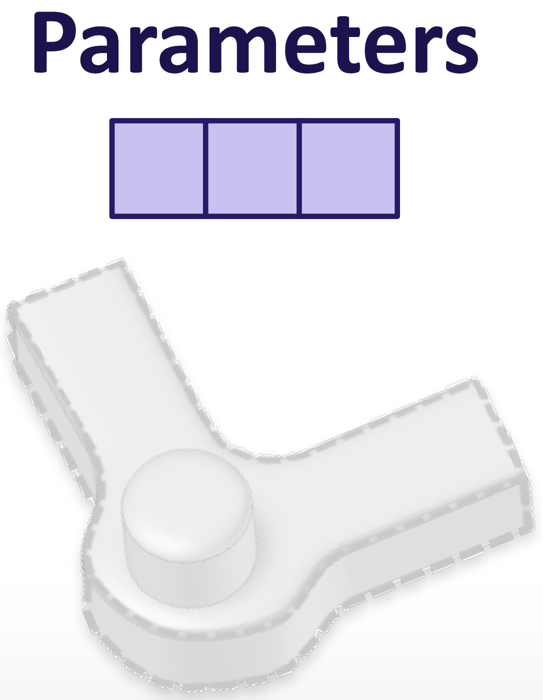
- Manual model setup

2.2 Fixed models

- Predefined variables

2.3 Data-driven parameters

- Discover and learn latent variables



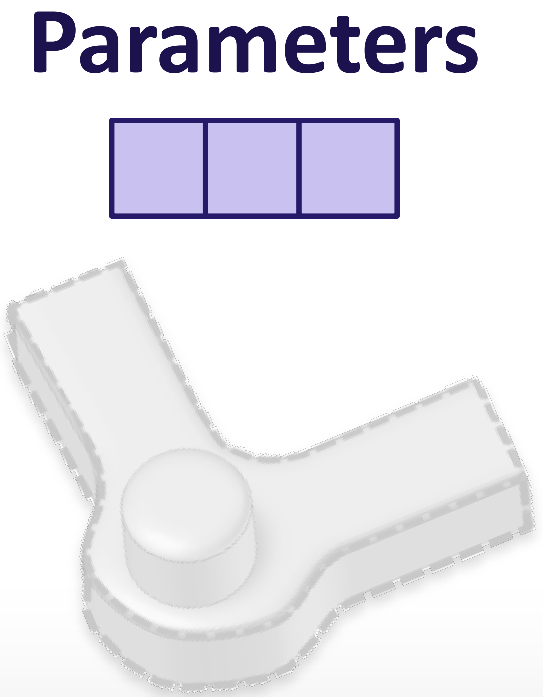
2. Parameters

Modeling and Detecting Parameters of Parts

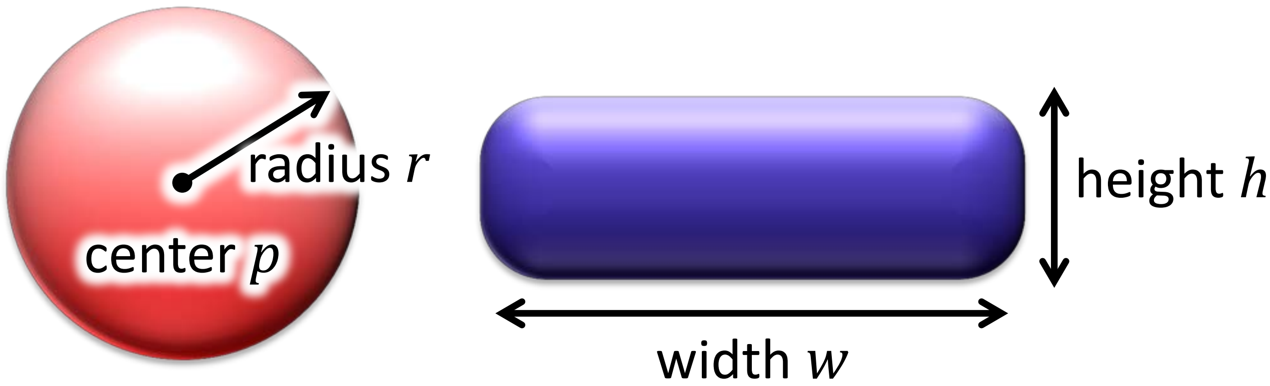
2.1 User defined parameters
- Manual model setup

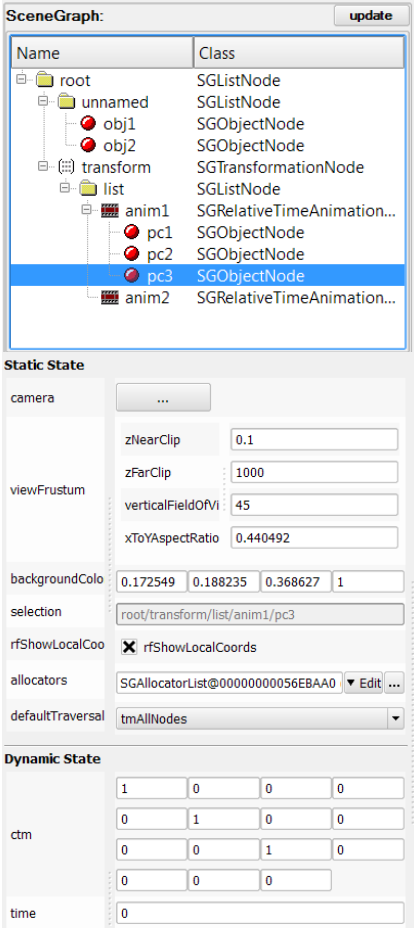
2.2 Fixed models
- Predefined variables

2.3 Data-driven parameters
- Discover and learn latent variables



User Defined Parameters





Traditional Modeling Systems

- Parameters of primitives
- Variables in scene hierarchies

Procedural Modeling

- Hierarchical model generation
- Scripts with local variables / arguments
  e.g. [Gervautz et al. 1996]

2. Parameters

Modeling and Detecting Parameters of Parts

2.1 User defined parameters

- Manual model setup

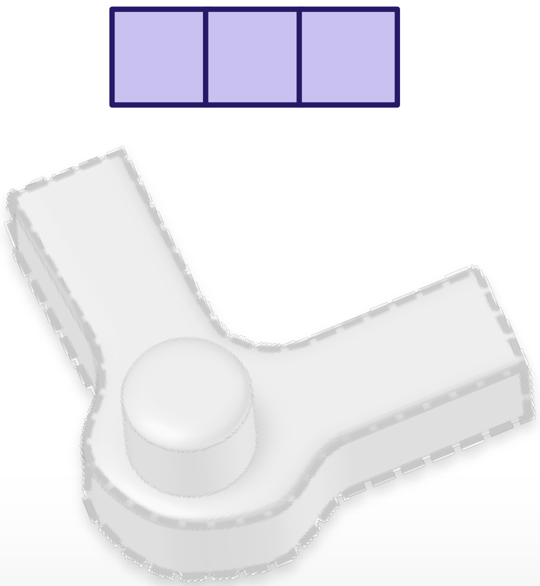
2.2 Fixed models

- Predefined variables

2.3 Data-driven parameters

- Discover and learn latent variables

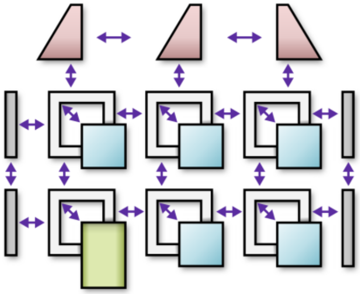
Parameters



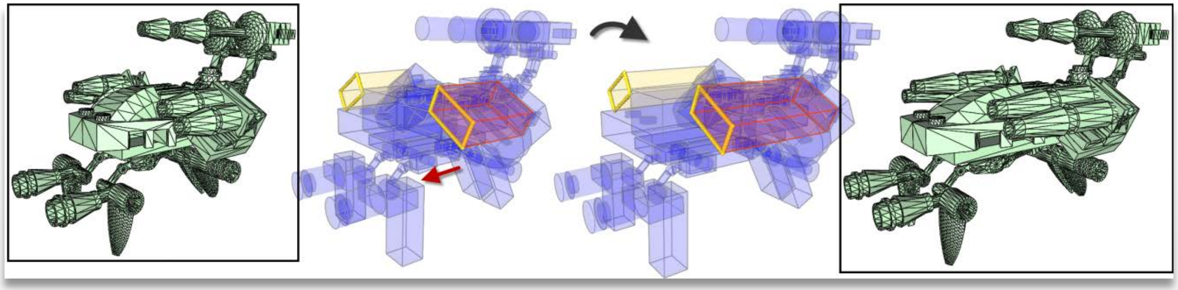
2.1 Fixed Parameter Models

Fixed Part Parameters

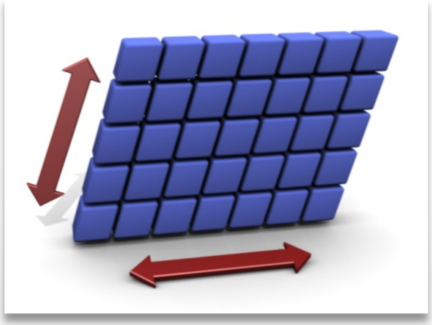
- Rigid/similarity-tr./affine mappings
- General shape spaces
  - Vertices move in $\mathbb{R}^{3n}$
- Constraint manifolds
  - Vertices move on smooth manifold $\mathcal{M} \subseteq \mathbb{R}^{3n}$
- Instantiation
- Controllers & Proxies



[Yang et al. 2011]



[Zheng et al. EG 2011]



[Bokeloh et al. SG 2012]

2. Parameters

Modeling and Detecting Parameters of Parts

2.1 User defined parameters

- Manual model setup

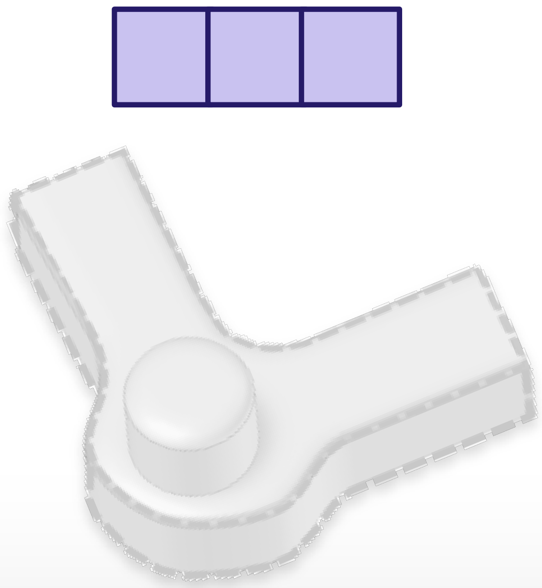
2.2 Fixed models

- Predefined variables

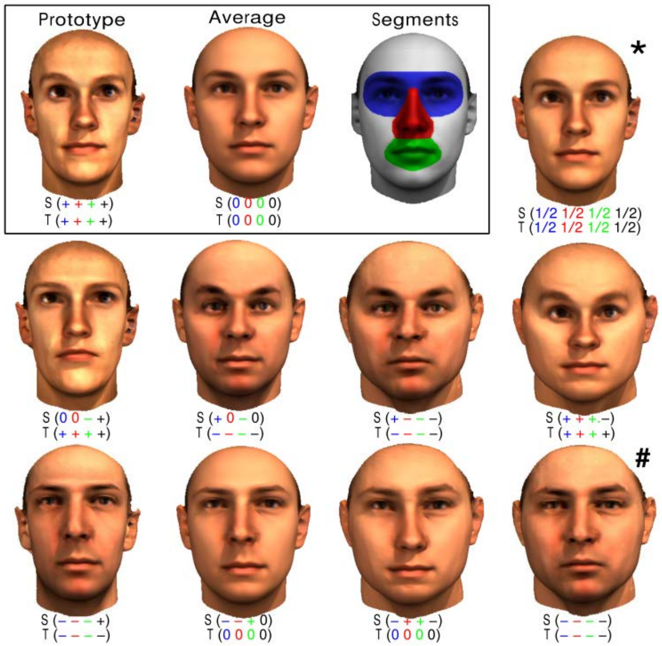
2.3 Data-driven parameters

- Discover and learn latent variables

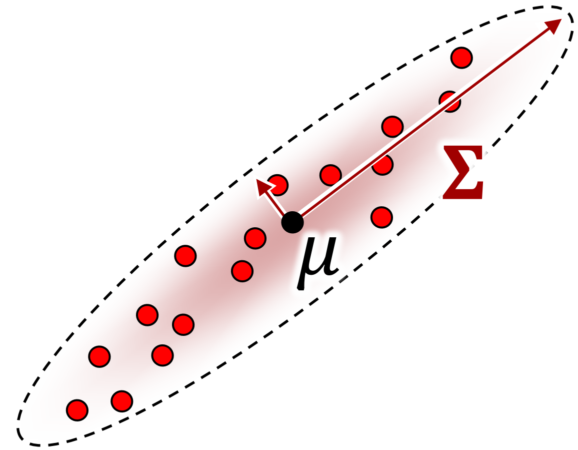
Parameters



Learning Parameters



[Blanz et al. 1999]



Learning Parameters from Data

- No prescribed variables / mappings
- Discover latent variables
- For example: Principal component analysis

Learning Parameters

Common Approaches

- Principal component analysis
  - "Morphable Models" [Cootes et al. 1998, Blanz et al. 1999, Allan et al. 2003, Anguelov et al. 2005, Hassler et al. 2009...]
- (Potentially) non-linear models
  - MDS / Kernel-PCA / spectral embedding

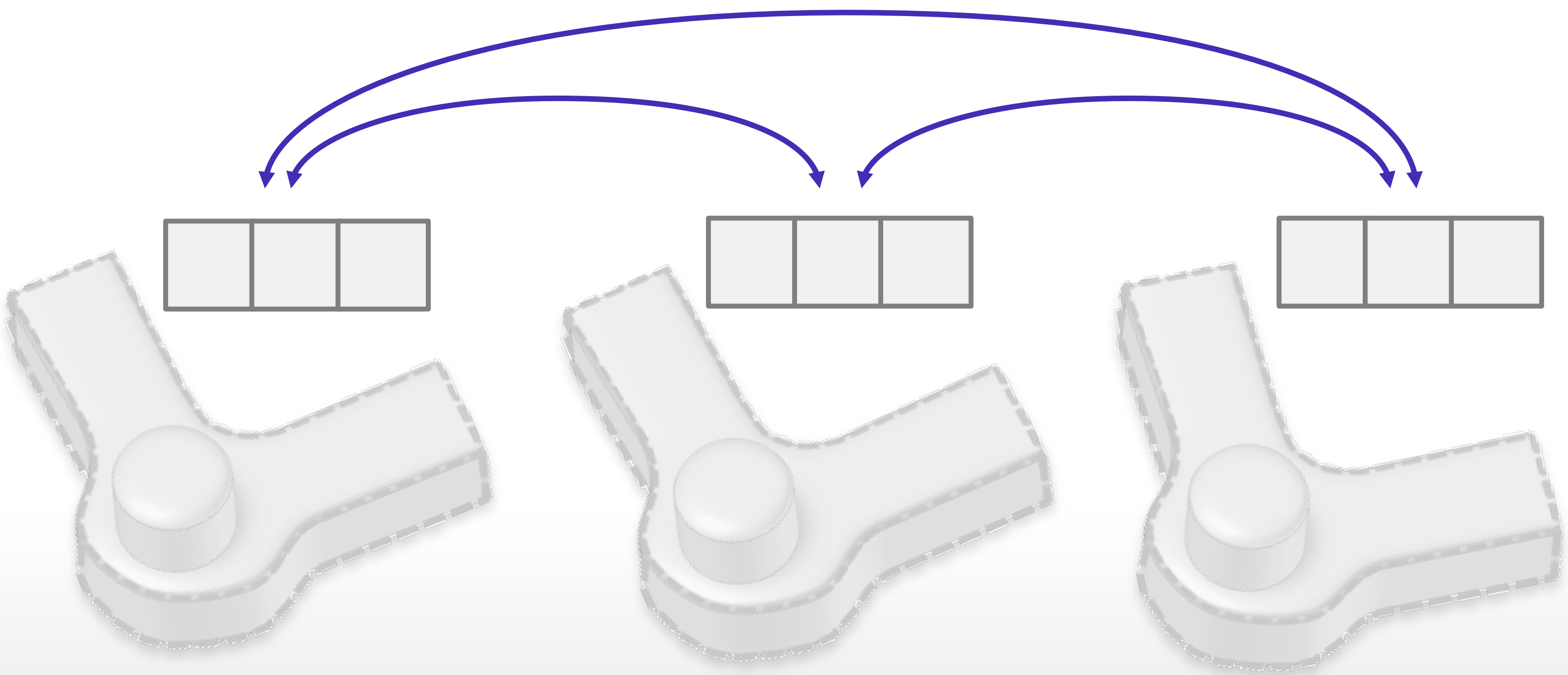
In general

- Latent variable models
- Examples
  - Independent component analysis
  - Latent variable Gaussian processes [Hertzmann et al. 2004]
  - ...

1. Parts

2. Parameters

3. Relations



3. Relations

Modeling and Detection of Relations

3.1 User defined relations

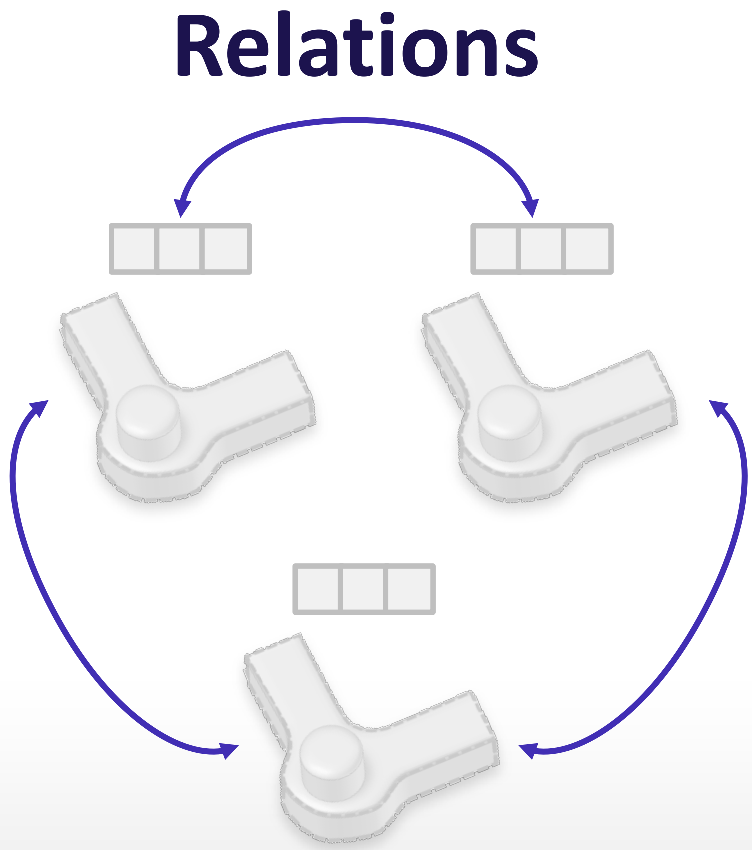
- Traditional constrained modeling

3.2 Fixed models

- Detect relations from predefined classes

3.3 Data-driven relations

- Discover new types of invariants from data



3. Relations

Modeling and Detection of Relations

3.1 User defined relations

- Traditional constrained modeling

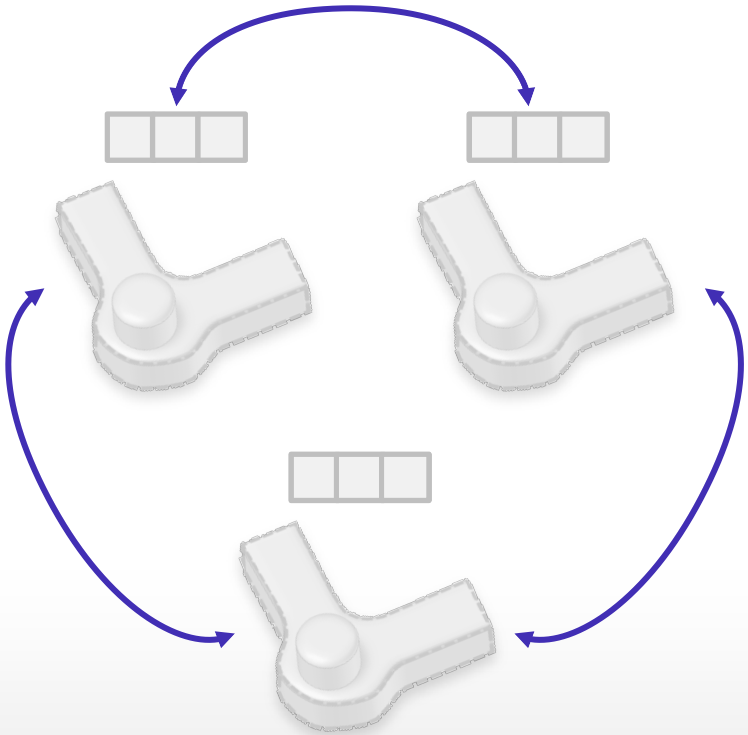
3.2 Fixed models

- Detect relations from predefined classes

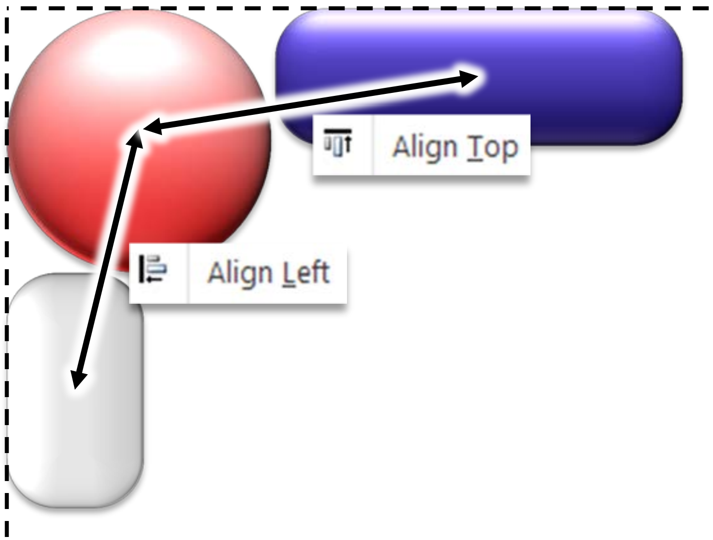
3.3 Data-driven relations

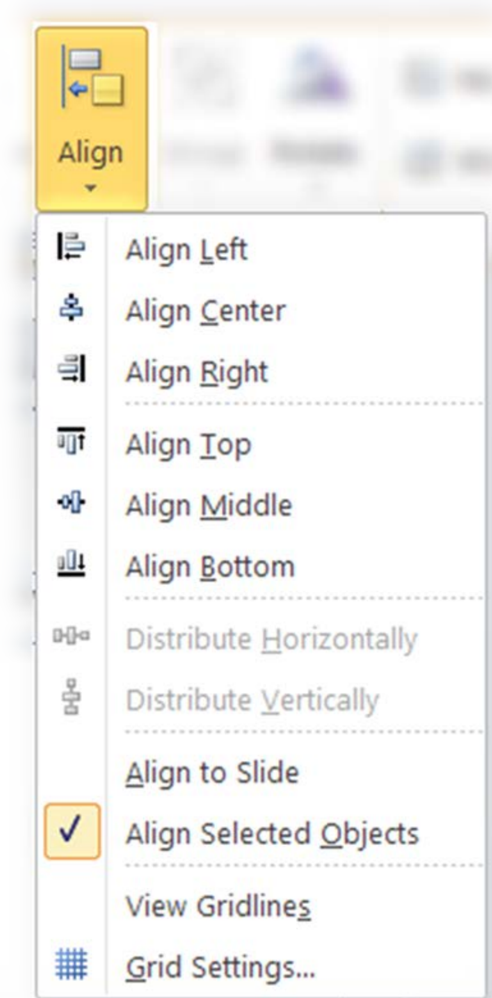
- Discover new types of invariants from data

Relations



3.1 Manual Relations





[MS Powerpoint 2010]

Constraints-Based Modeling

- Specify desired constraints
  - Part parameters
  - Derived properties
- Constraints-solver (hard/soft)

3. Relations

Modeling and Detection of Relations

3.1 User defined relations

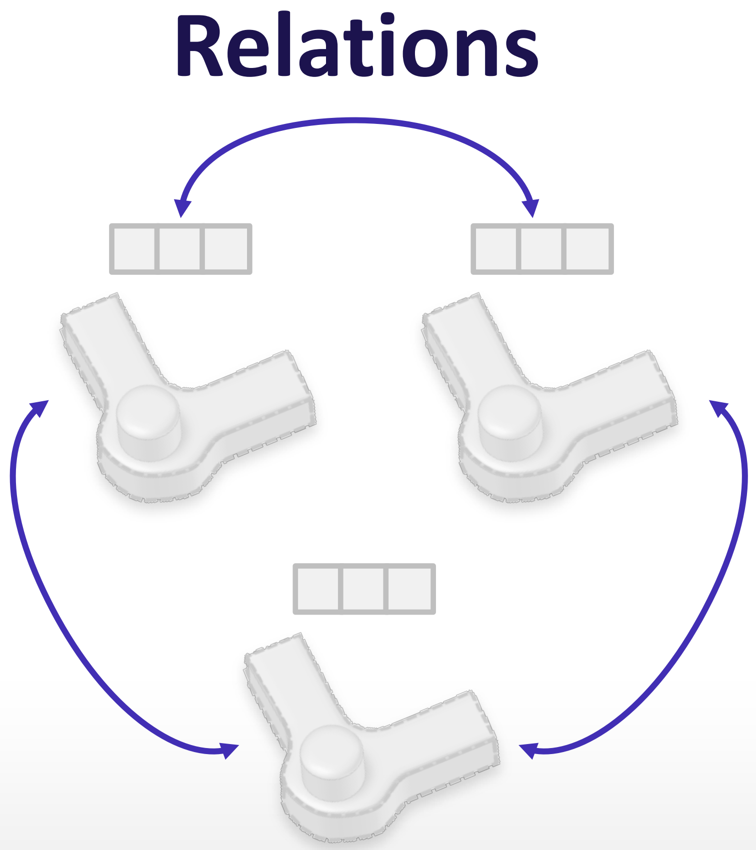
- Traditional constrained modeling

3.2 Fixed models

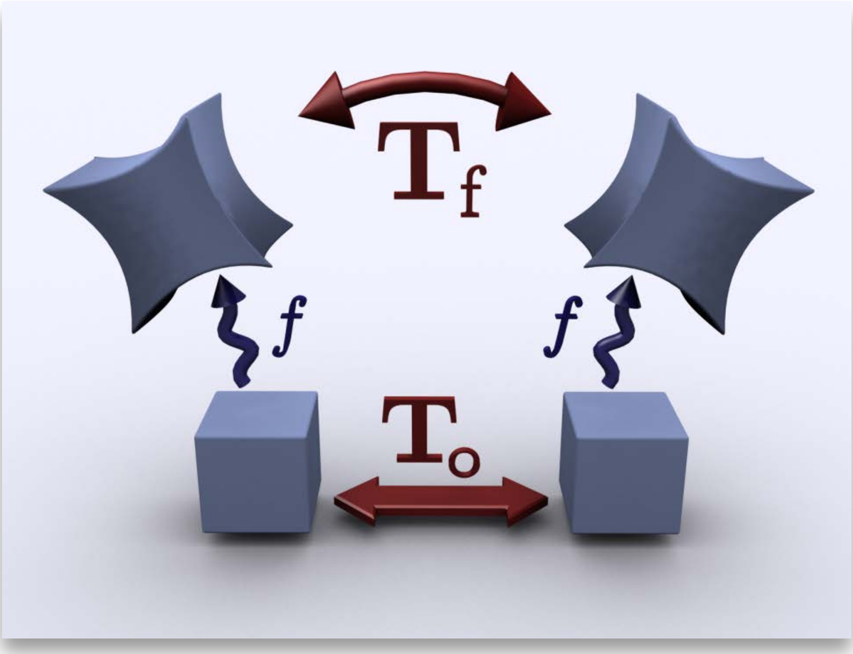
- Detect relations from predefined classes

3.3 Data-driven relations

- Discover new types of invariants from data



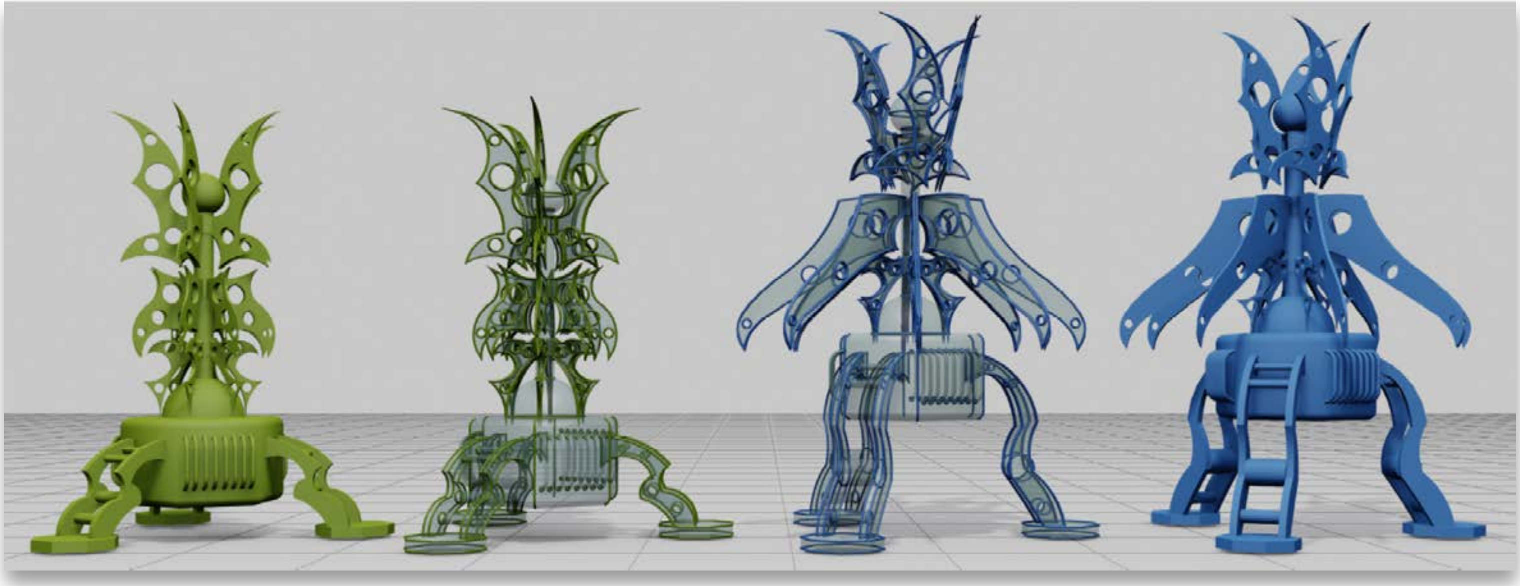
3.2 Predefined Classes of Relations



Symmetry and Euclidean Invariants

- Detect symmetries
  - Also: partial Euclidean invariants (parallelity, distances, angles, etc.)
- Build constraints (soft/hard)

3.2 Predefined Classes of Relations



[Gal et al. SG 2009]

Example System: iWires

- Detection:
  - Feature-based approach
  - Line features (“wires”)

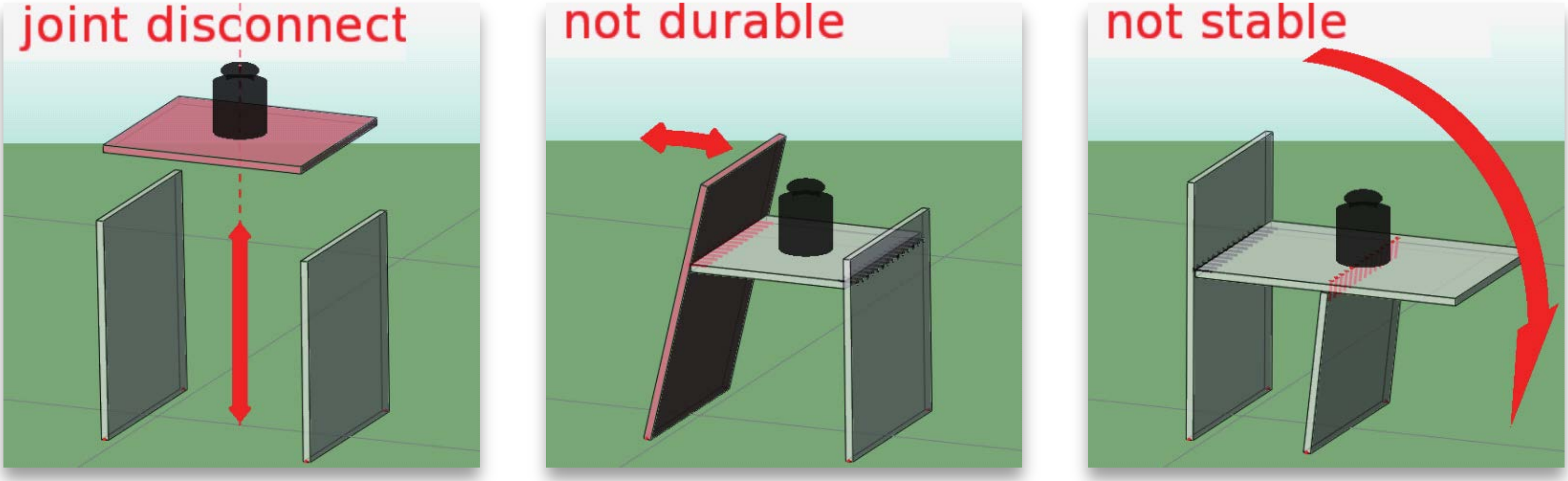
Physical Constraints

Physics

- Stability
  - Stable standing under gravity
  - Ability to withstand forces in use
- Functionality
  - Transmission of forces, movements (rotational, translational)
  - Movability
- Assembly
  - Parts can be put together
  - Accessibility (service)
- etc...

Potential for future work

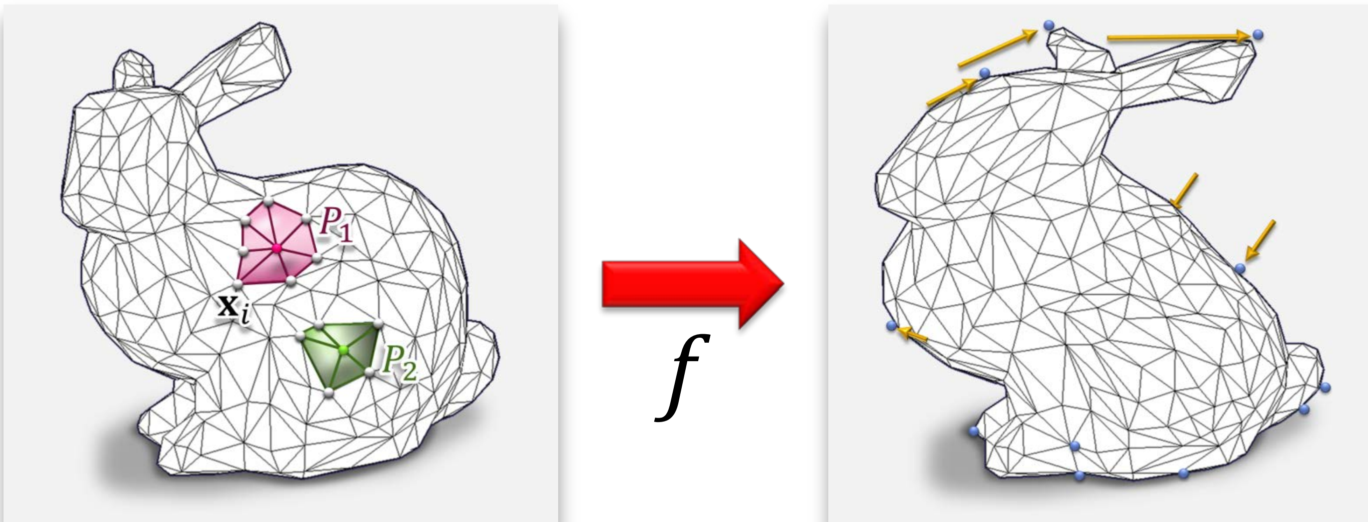
Physical Constraints



[Umetani et al. SG 2012]

Example: Stability of Furniture

Low-Level Physics



Example: Deformation models – soft constraints

- Elastic deformation

$$E(f) = \int_\Omega \left\| \nabla f(x) \cdot [\nabla f(x)]^{\mathrm{T}} - \mathbf{I} \right\|_F^2 dx$$

- Thin-plate splines

$$E(f) = \int_\Omega \left\| \mathbf{H}_f \right\|_F^2 dx$$

3. Relations

Modeling and Detection of Relations

3.1 User defined relations

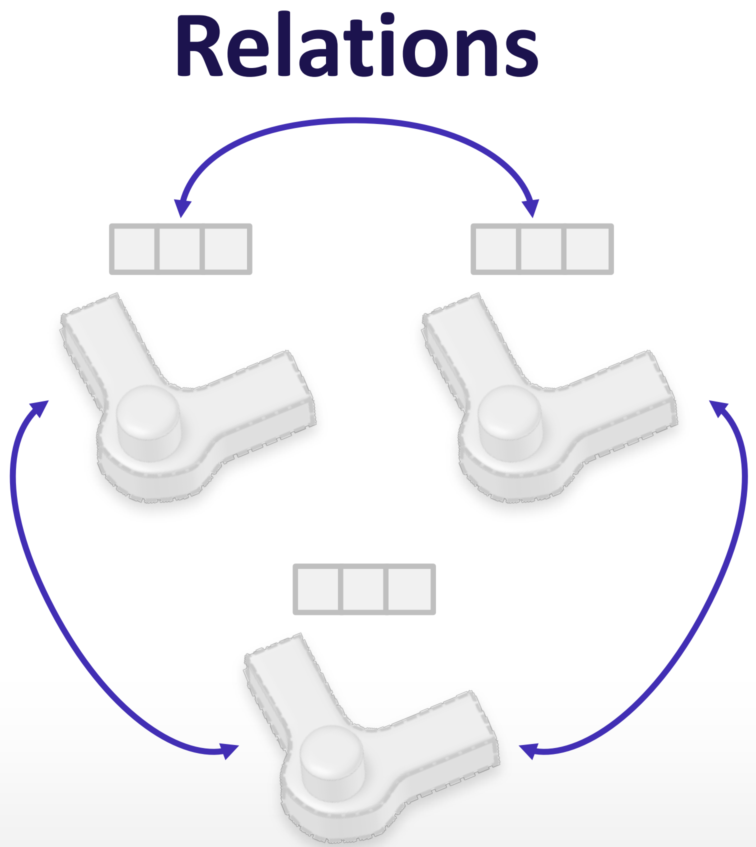
- Traditional constrained modeling

3.2 Fixed models

- Detect relations from predefined classes

3.3 Data-driven relations

- Discover new types of invariants from data



Learning Constraints

Key problem

- Establish correspondences between parts & parameters

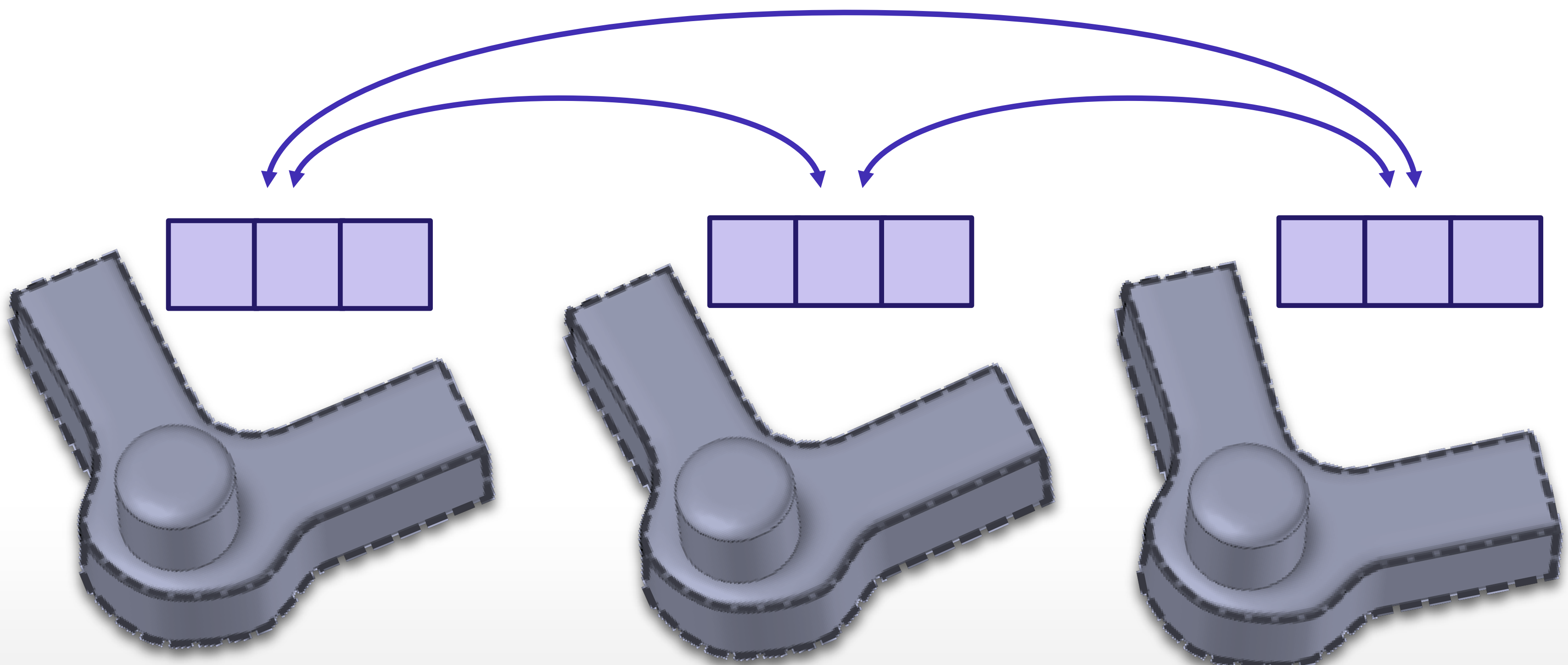
Latent parameter models

- Dimensionality reduction (PCA),
  manifold learning (kernel-PCA, spectral embeddings)

Shape assemblies

- Learning scene layout [Fisher et al. 2010-2012]
- Learning how to assemble shapes [Kalogerakis et al. 2012]

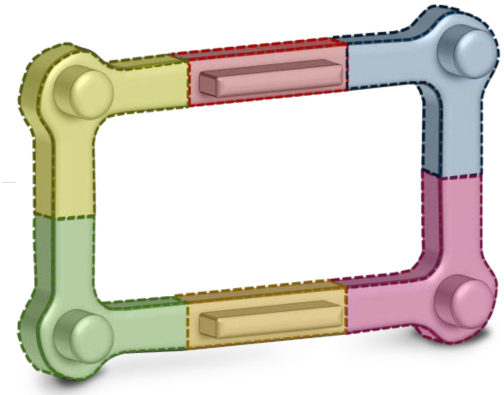
Summary

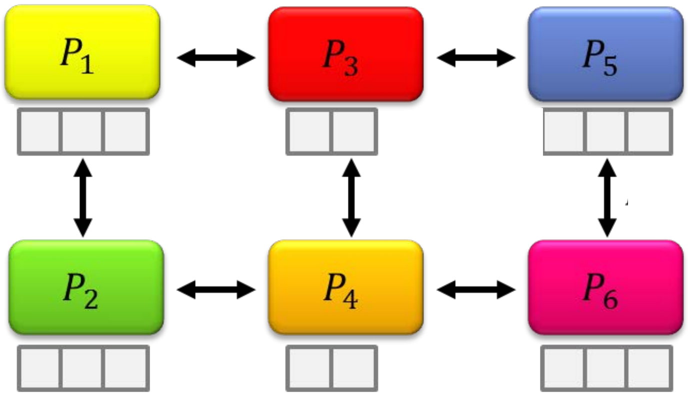


Structure Models & Analysis

Types of Structure

- Model components
  - Parts
  - Parameters
  - Relations
- Classification
  - User input
  - Fixed models
  - Data-driven models





Major challenge

- Towards data-driven methods, based only on low-level assumptions?