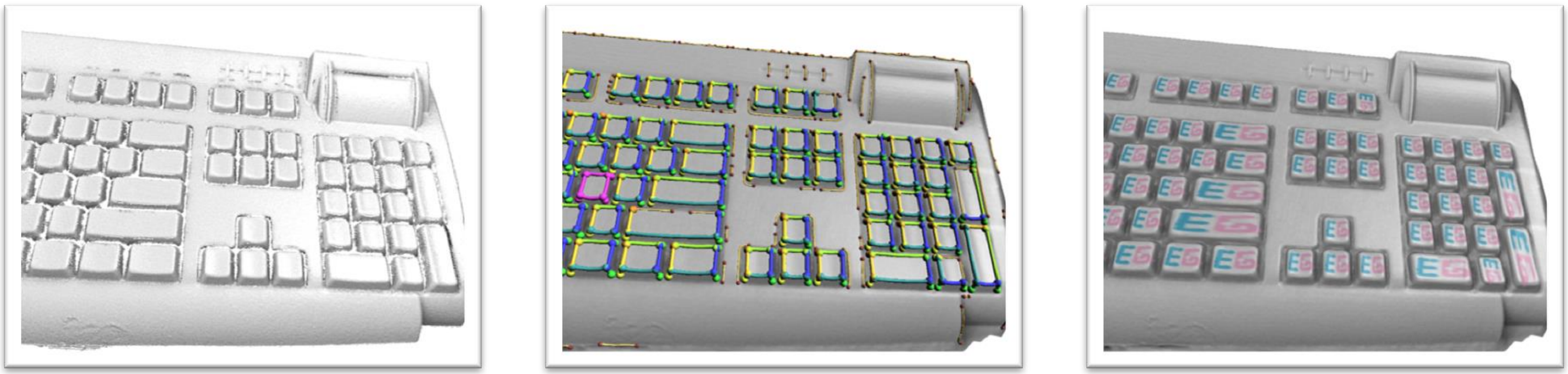
Shape Analysis with Subspace Symmetries



Alexander Berner[1] **Michael Wand**[1,2] **Niloy Mitra**[3,4]

Daniel Mewes[1] **Hans-Peter Seidel**[1]

[1]MPI Informatik [2]Saarland University [3]KAUST [4]IIT Delhi

Eurographics 2011
LLANDUDNO UK | Bangor University School of Computer Science
11-15 April 2011

max planck institut informatik

UNIVERSITÄT DES SAARLANDES

CLUSTER OF EXCELLENCE

iitdelhi

Overview

Introduction

Related Work

Subspace Symmetries

Detection Algorithm

Results

Conclusions

Overview

Introduction

Related Work

Subspace Symmetries

Detection Algorithm

Results

Conclusions

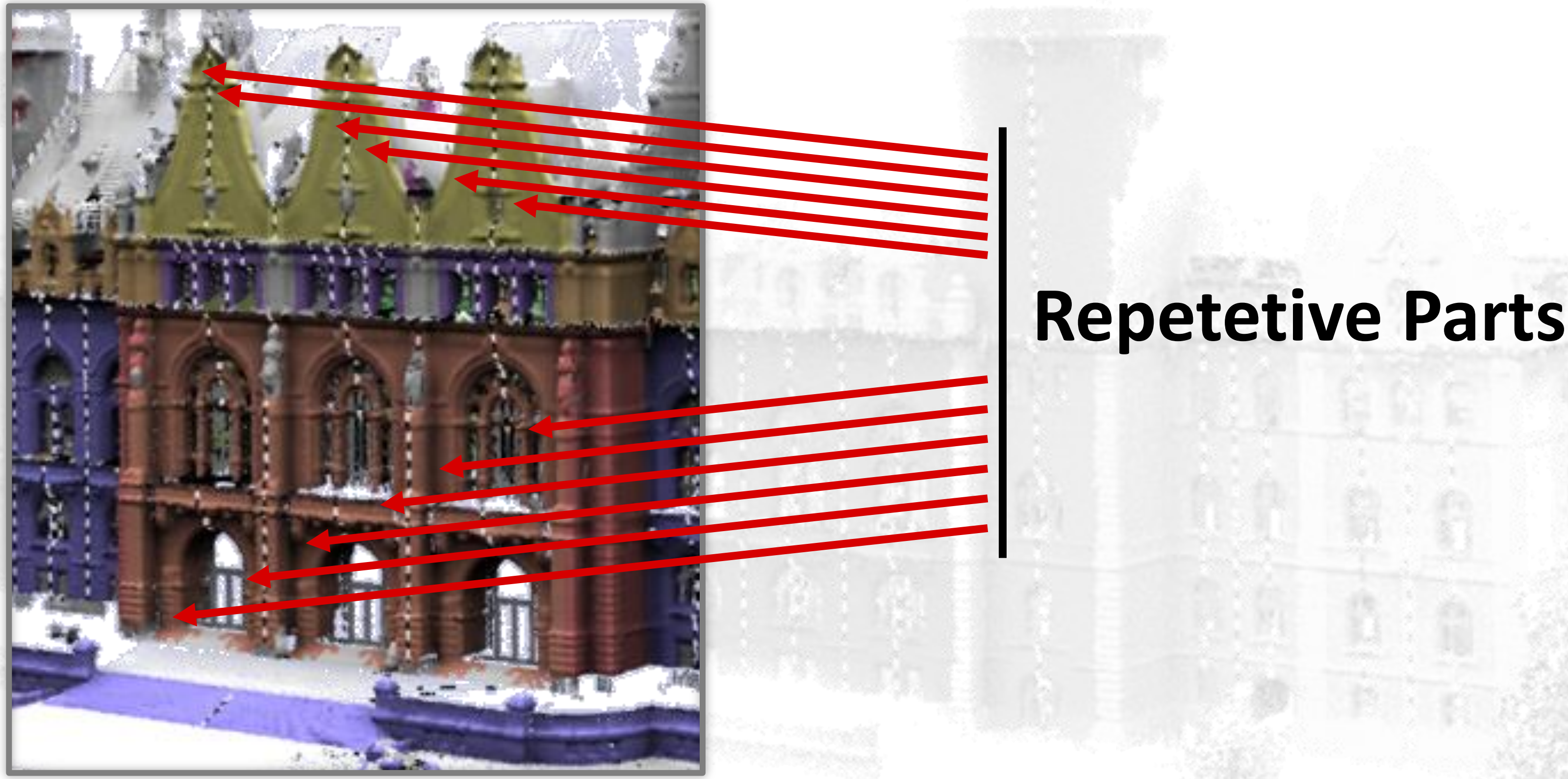
Partial Symmetries

Partial Symmetry Detection

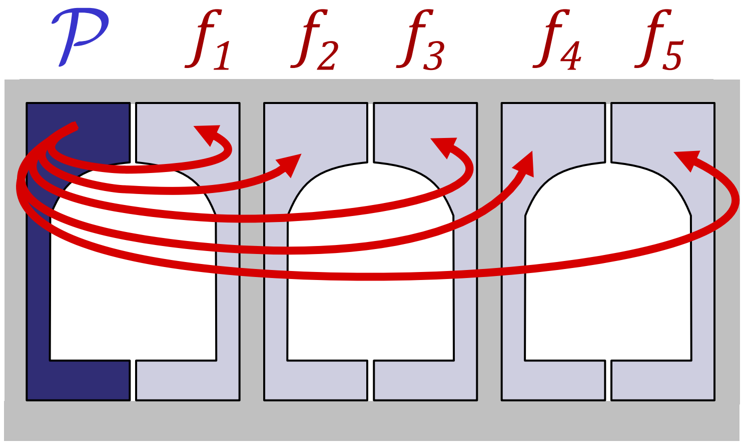
- Find similar parts
- Decomposition into building blocks
- Fundamental tool in shape understanding

Partial Symmetries

Partial Symmetry Detection



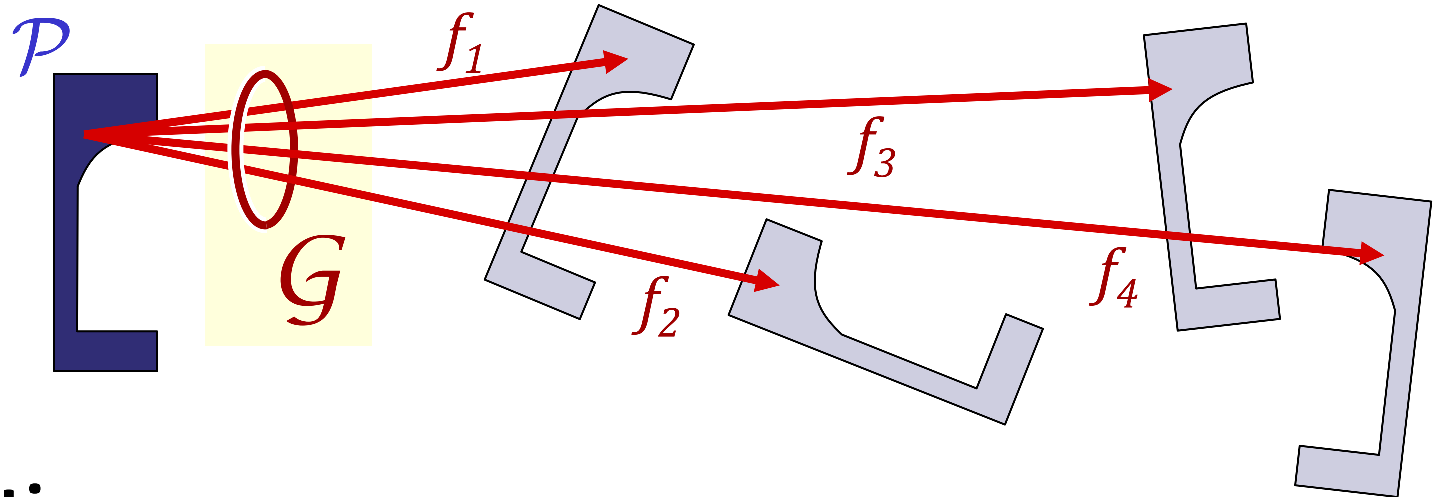
Partial Symmetries



Partial Symmetry Detection

- Repetitive part $\mathcal{P}$ (sufficiently large)
- Transformations $f_i \in \mathcal{G}$
- Group of transformations $\mathcal{G}$

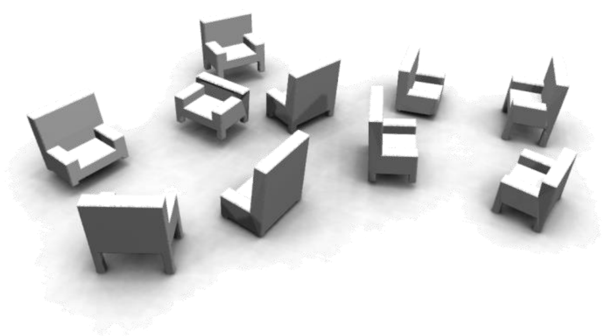
Restriction

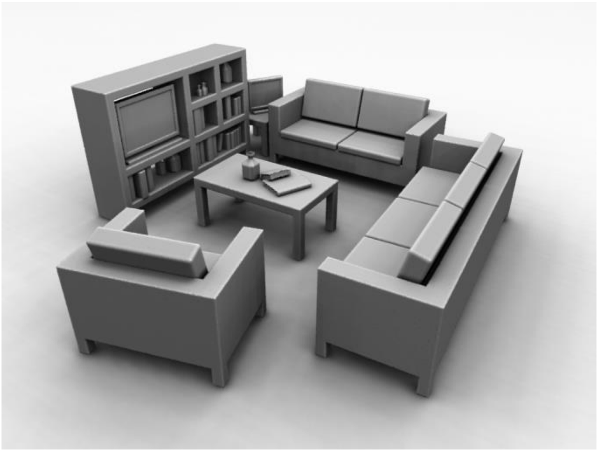


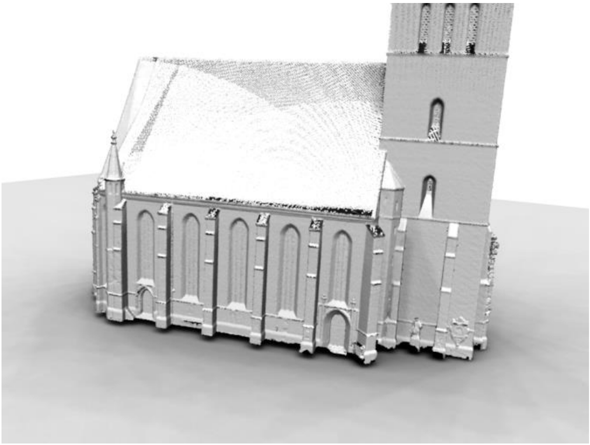
Restriction

- Fixed group of transformations
  - Rigid motions, reflections, scaling, affine maps
  - Intrinsic isometries
- Need to define *a priori* what constitutes similarity

More General Symmetries







Overview

Introduction

Related Work

Subspace Symmetries

Detection Algorithm

Results

Conclusions

Symmetry Detection

Fixed Transformation groups

- Reflections

  [Podolak et al. 2006], [Loy et al. 2006]

- Euclidean Transformations

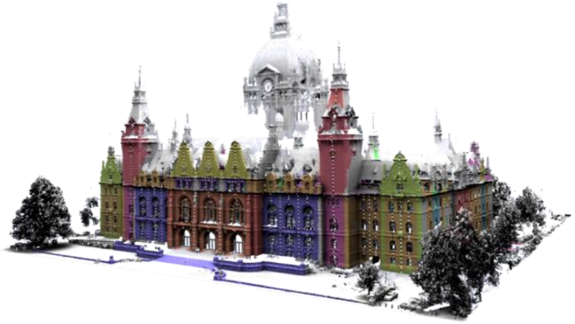
  [Bokeloh et al. 2009]

- Similarity transforms

  [Mitra et al. 2006], [Pauly et al. 2008]

- Intrinsic isometries

  [Ovsjanikov et al. 2008],
  [Lasowski et al. 2009], [Xu et al. 2009]
  [Mitra et al. 2010], [Kim et al. 2010]



Global Matching of General Shapes

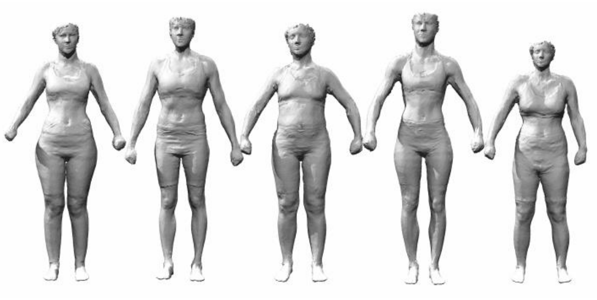
Global Matching

- Topological Methods
  - [Hilga et al. 2001]
- Combinatorial Search
  - [Zhang et al. 2008], [Au et al. 2010]
- Learning
  - [Kalogerakis et al. 2010],
    [van Kaik et al. 2011], [Sunkel et al. 2011]

Global Matching of General Shapes

Building subspace models

- Local matching, user guided
  - [Blanz et al. 1999], [Allen et al. 2003], [Hasler et al. 2009]



Overview

Introduction

Related Work

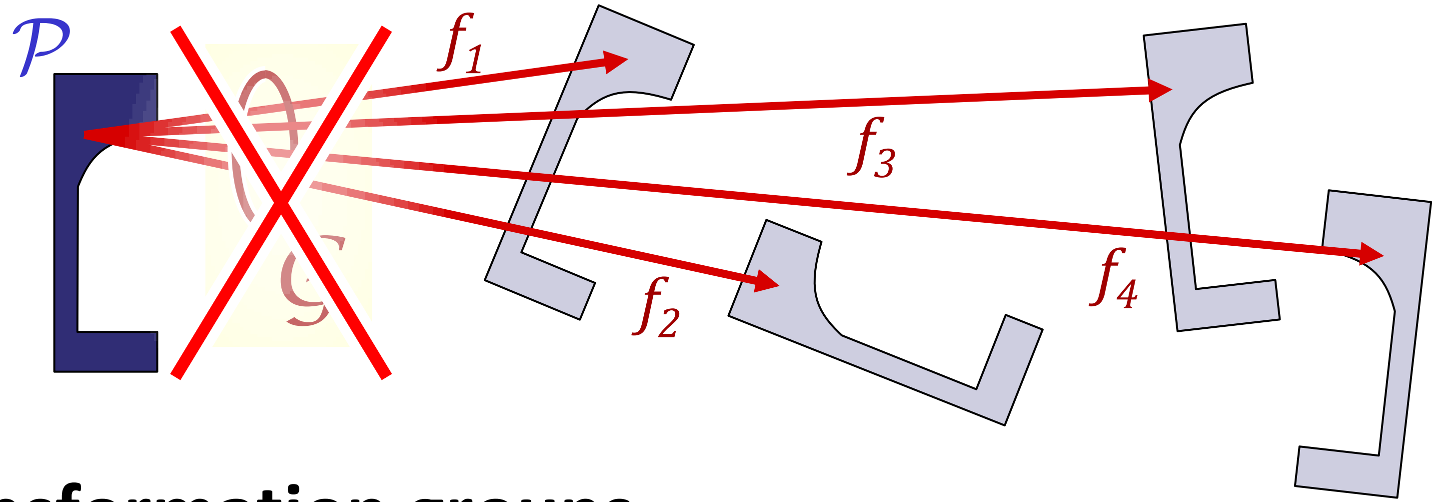
Subspace Symmetries

Detection Algorithm

Results

Conclusions

Subspace Symmetries



No transformation groups

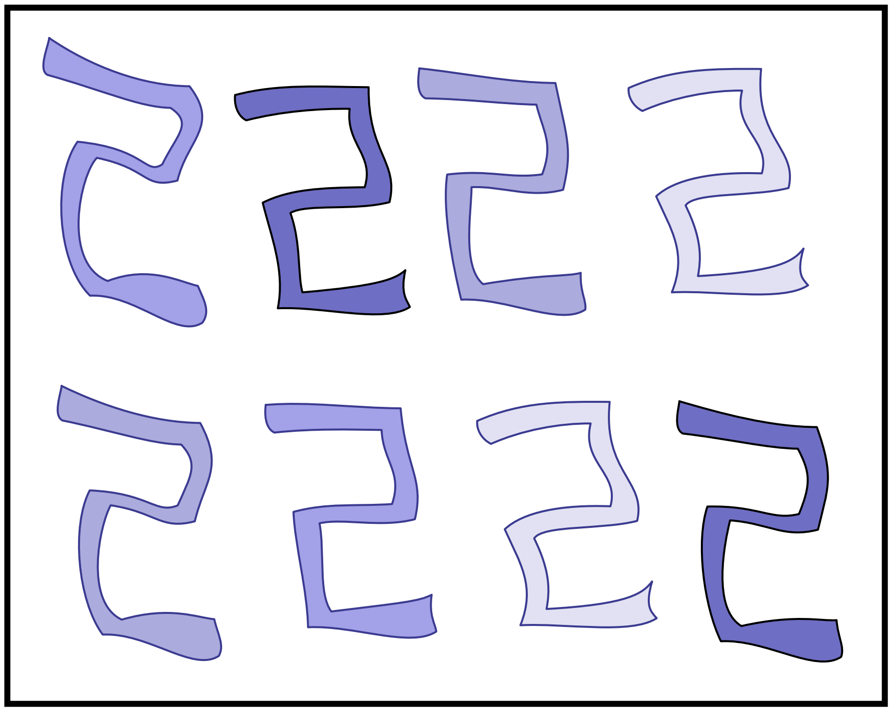
- (Almost) arbitrary mappings
- How to avoid spurious matches?

Key idea

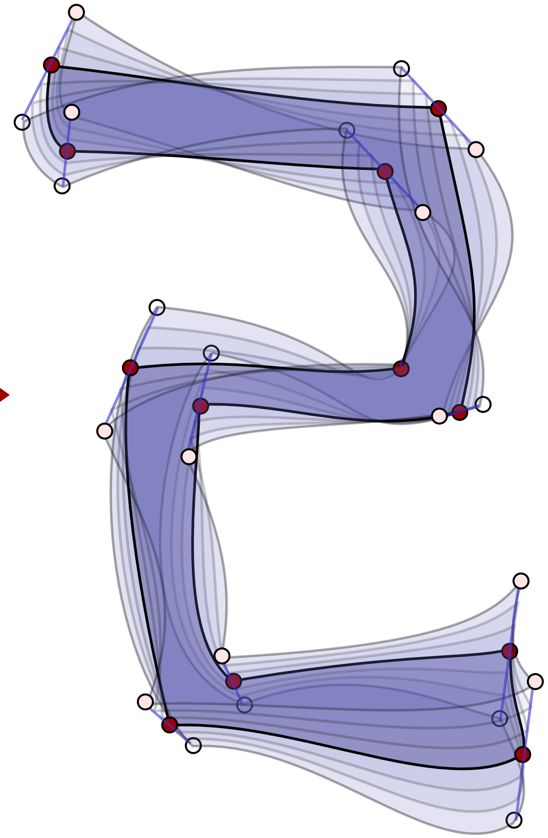
- Matching functions must form low dimensional subspace

Subspace Symmetries

Shapes



Subspace (1D)



Objective

Find

- Part $\mathcal{P}$
- Functions $f_1, \ldots, f_n$

Such that:

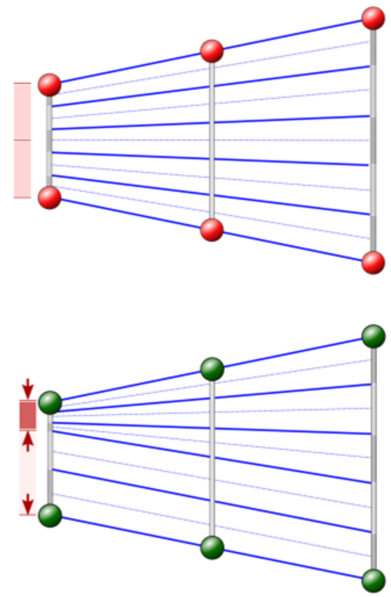
$$f_i(\mathcal{P}) = \overbrace{\mathbf{T}}^{\substack{\text{Rigid Motion} \\ \text{(Param)}}} \left(\underbrace{\mathcal{P}}_{\substack{\text{Mean} \\ \text{(Model)}}} + \sum_{k=1}^{d} \overbrace{\lambda_k}^{\substack{\text{Shape Coordinates} \\ \text{(Param)}}} \underbrace{b_k(\mathcal{P})}_{\substack{\text{Basis function} \\ \text{(Model)}}} \right)$$

$$d \ll n$$

Remarks

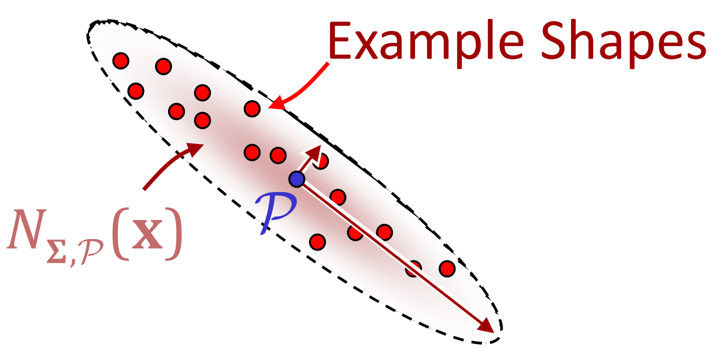
Uniqueness:

- Many aquivalent subspace models might fit the same data
- Symmetry breaking: minimize bending



Gaussian Model:

- We can learn covariance from data
- Additional constraint



Challenge

Input

- Shape $\mathcal{S} \subseteq \mathbb{R}^3$

Unknowns

- Part $\mathcal{P} \subseteq \mathcal{S}$
- Functions $f_1, \ldots, f_n$

Can be computed

- Rigid transformations $\mathbf{T}_1, \ldots, \mathbf{T}_n$
- Basis functions $b_1, \ldots, b_n$
- Shape coordinates $\lambda_1, \ldots, \lambda_n$

Rigid Motion (Param)

Shape Coordinates (Param)

$$f_i(\mathcal{P}) = \mathbf{T}\left(\mathcal{P} + \sum_{k=1}^{d} \lambda_k b_k(\mathcal{P})\right)$$

$d \ll n$

Mean (Model)

Basis function (Model)

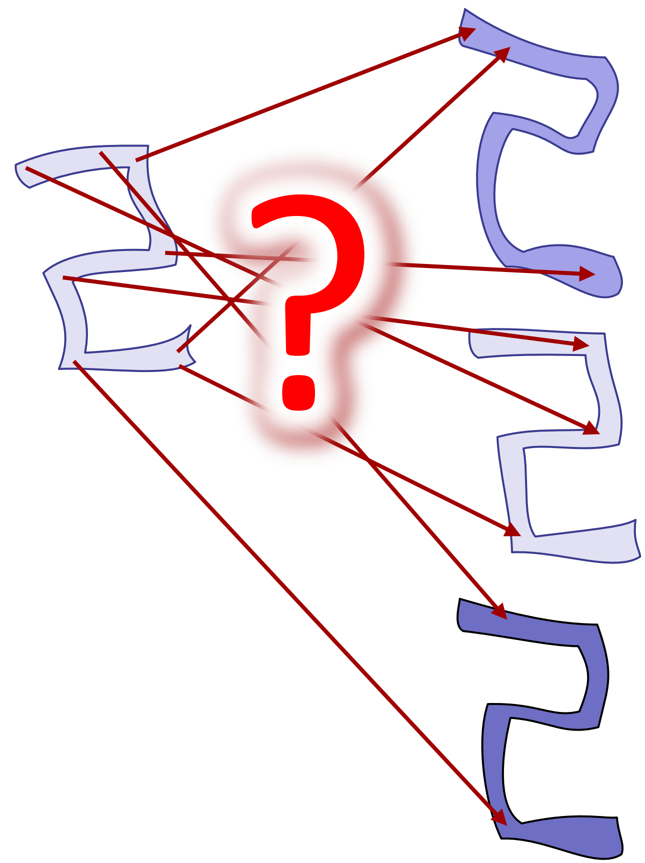
Challenge

Unknowns

- Part $\mathcal{S} \subseteq$ ☼^{3}
- Functions $f_1, \ldots, f_n$

Problems

- Need correspondences
- High dimensional objects
- Very large search space



Overview

Introduction

Related Work

Subspace Symmetries

Detection Algorithm

Results

Conclusions

Three Steps to Reduce Complexity

1. Feature matching

- Sparse, discrete matching

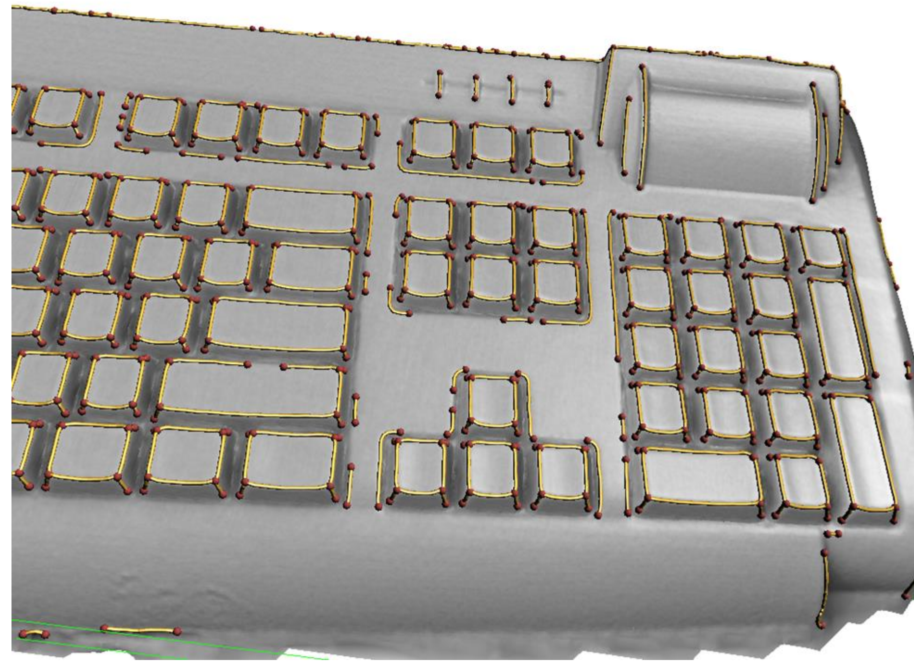
2. Graph matching

- Matching heuristic

3. Optionally: User training

- Learn graphs from user input

Feature Extraction



Features: surface curves & crossings

- **Strong assumption:** Graphs invariant under symmetry
- See paper technical details

Feature Matching

Brute-foce feature matching

- *d*-dimensional subspace, *n* feature points
- Brute force algorithm: double exponential in *d*

Need more efficient strategy

Heuristic Bootstrapping

Stronger Assumption

- Corresponding parts have *similar feature graphs*

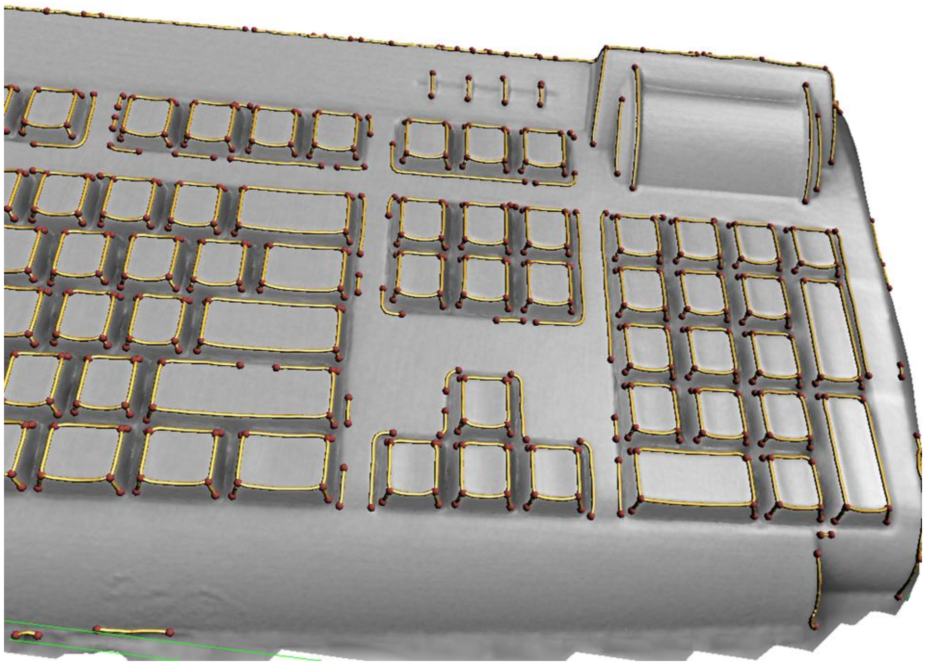
Similar

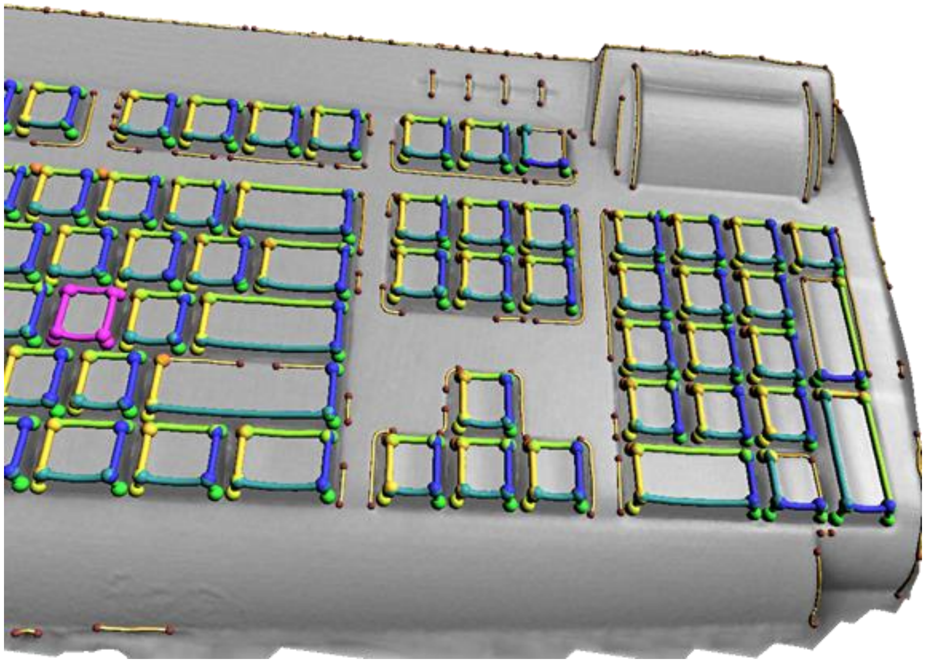
- Same topology (small defects possible)
- Similar geometry
  - Angles, up to some noise
  - Intrinsic distances up to factor 3x

Bootstrapping

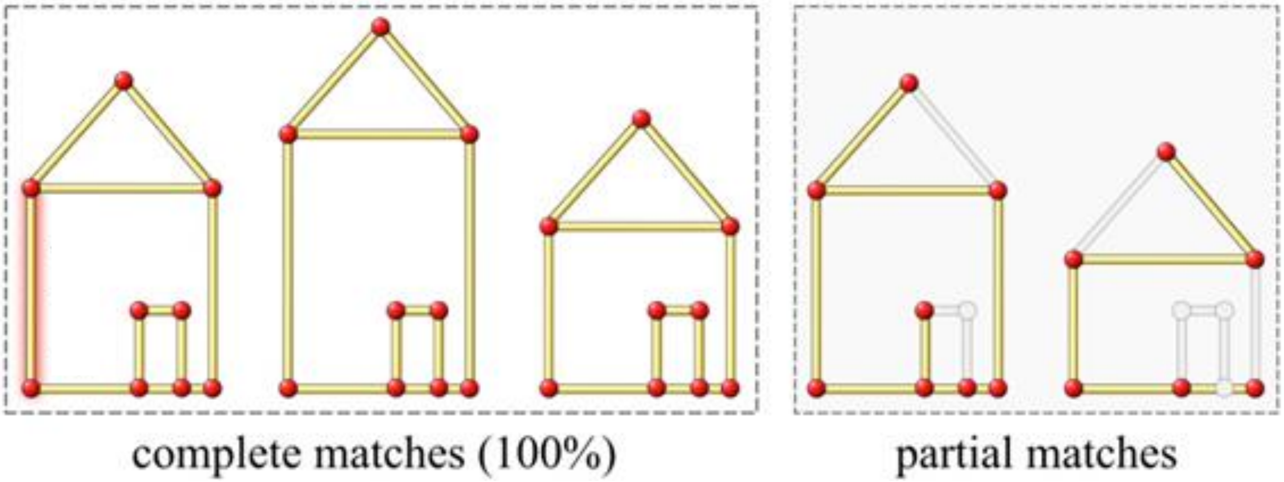
- Find a few instances first, build PCA model
- Partial finds more

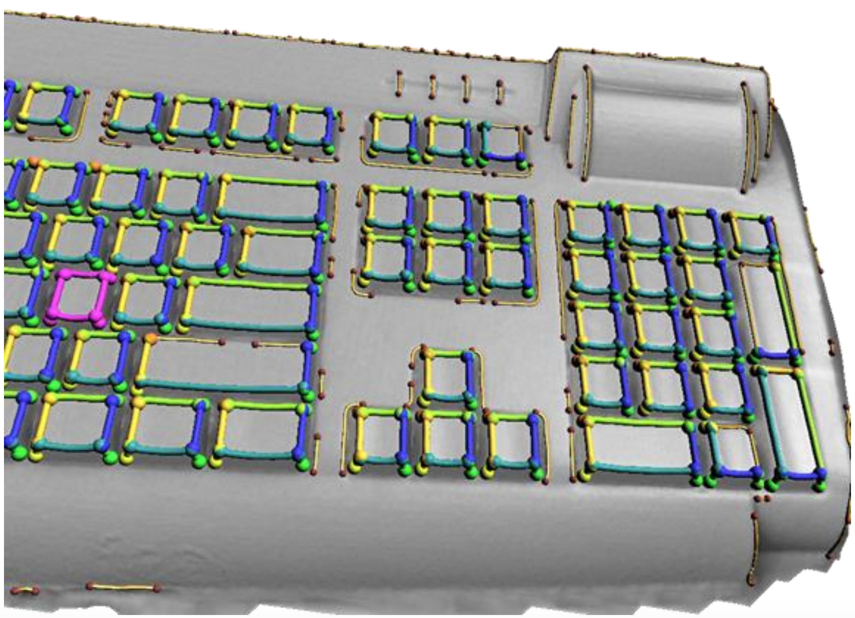
Graph Matching

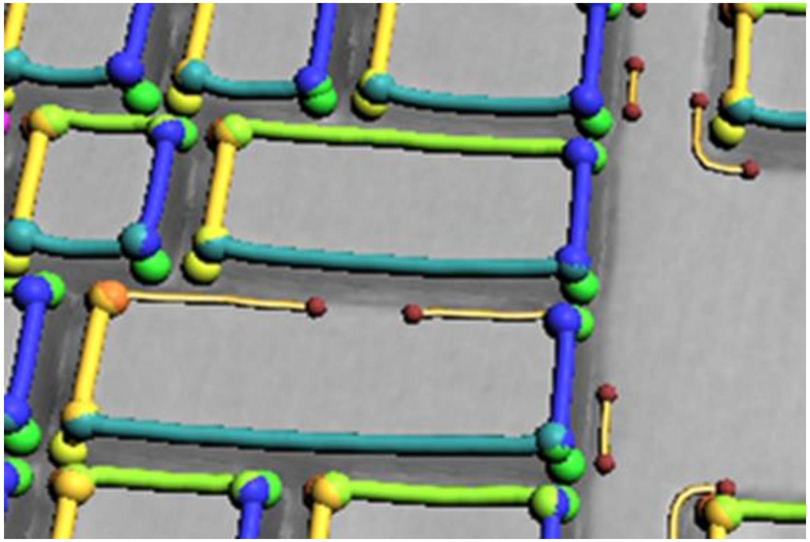




Complete & Partial Matches

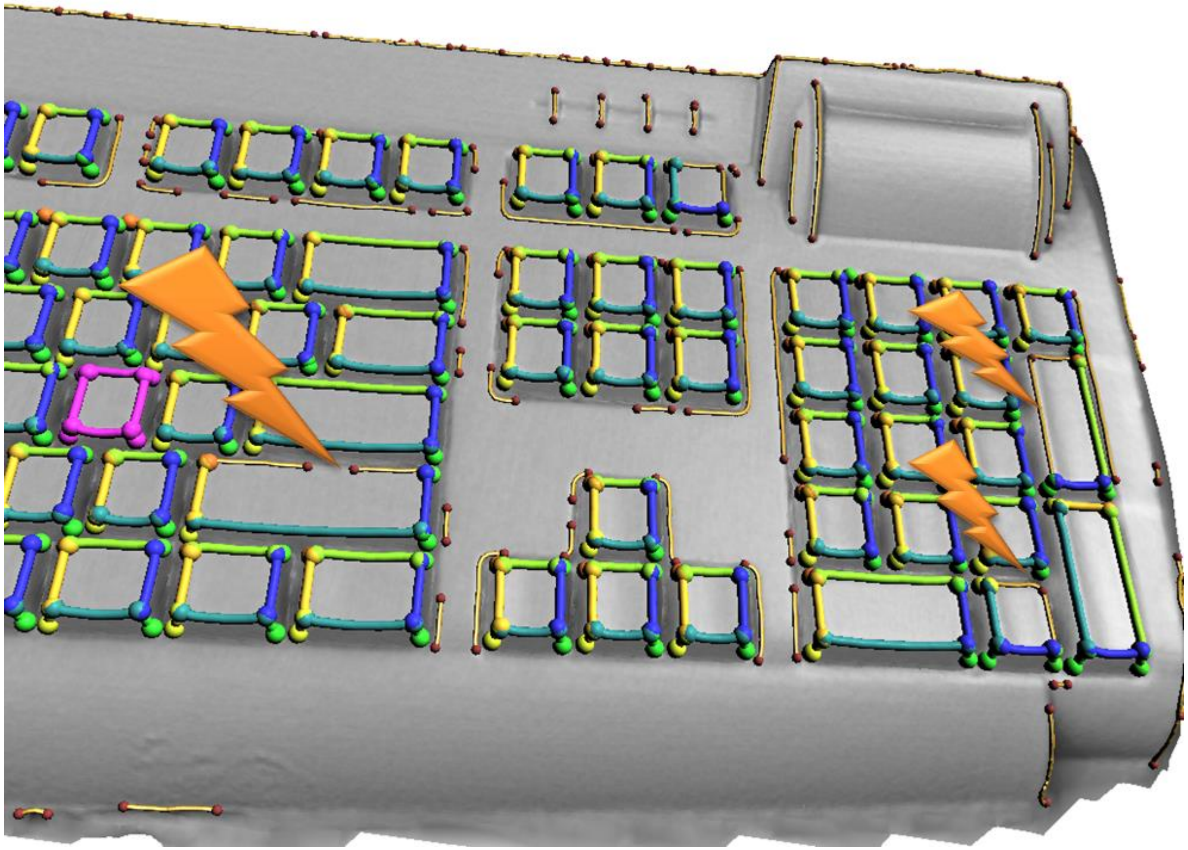


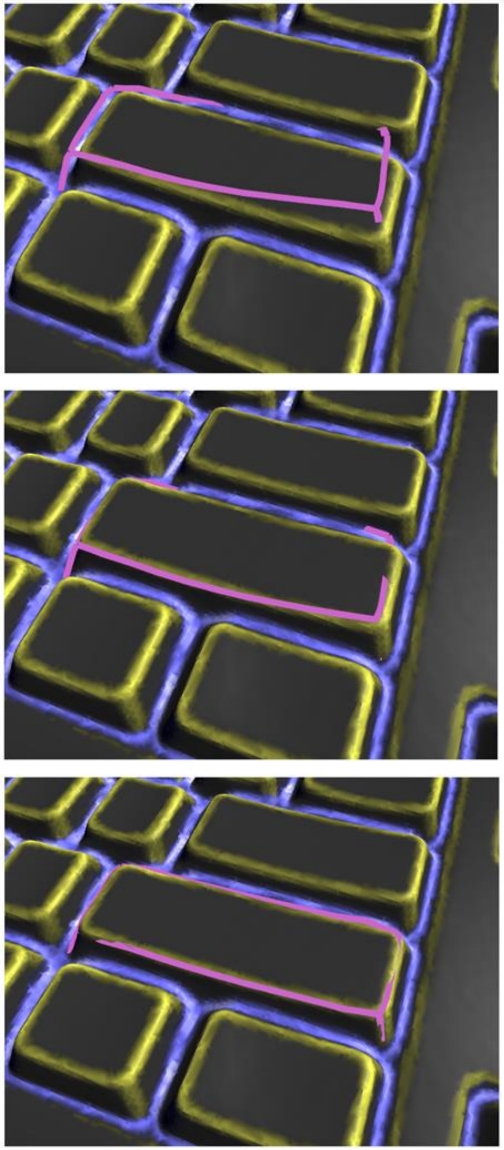




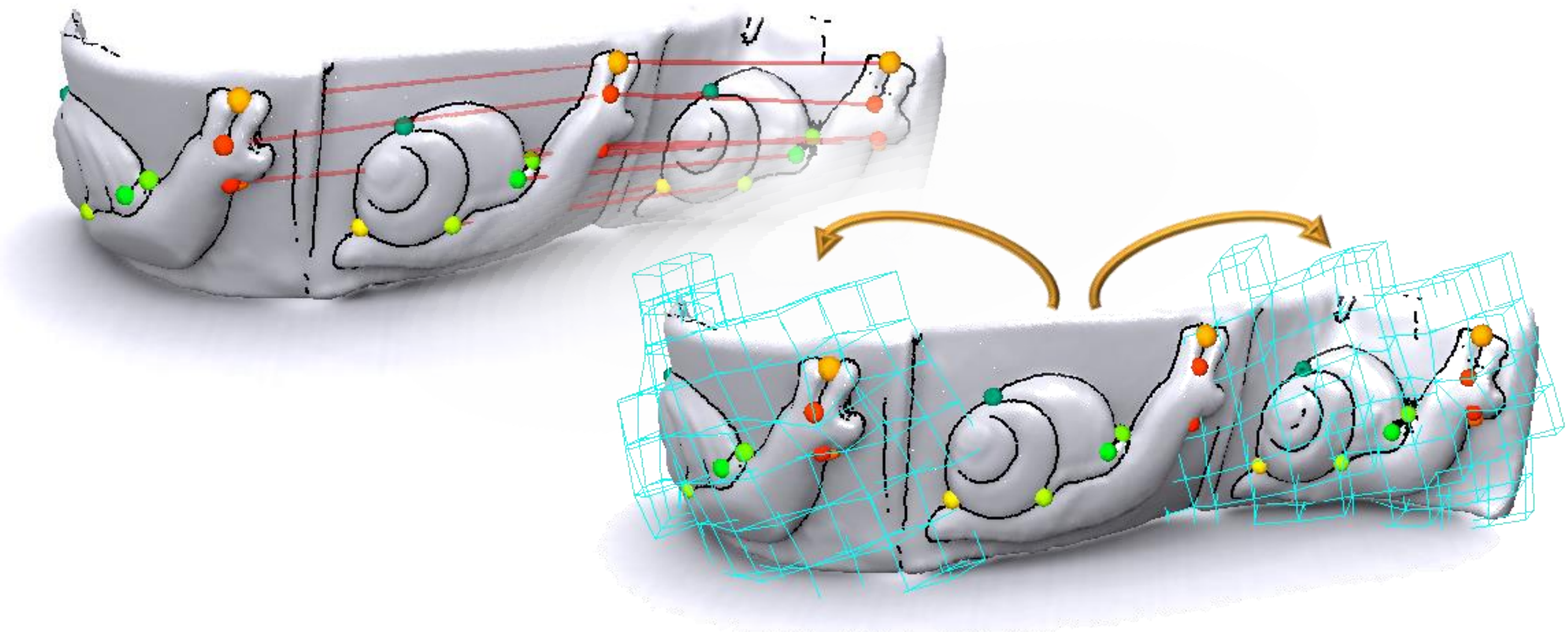
Refinement

Use discovered subspace model





Dense Correspondences

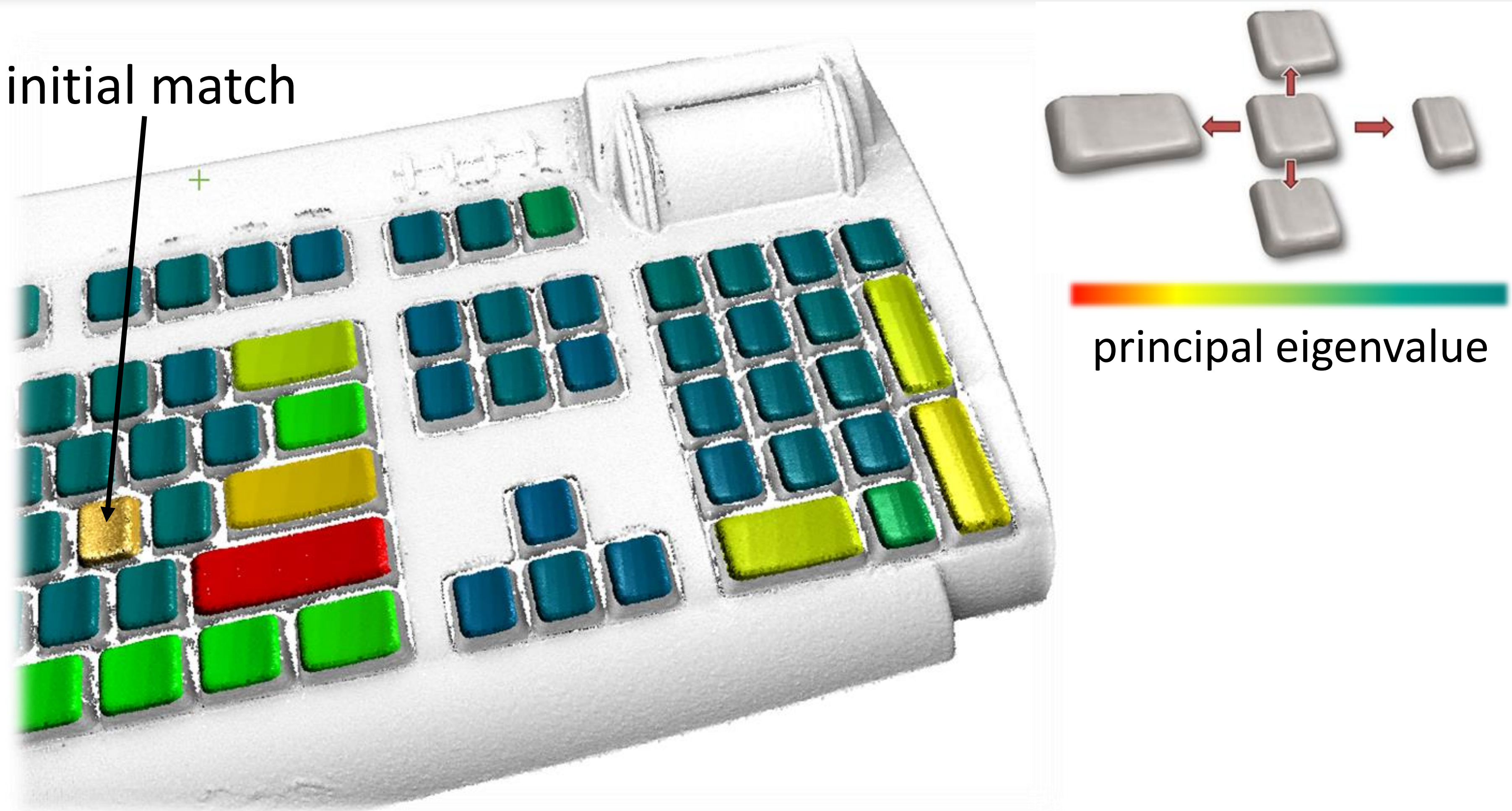


Deformable ICP

- Fit bending minimizing dense correspondence field
- Thin-plate-splines

Result

Result



Extension: Manual Training

Training

- Click on corresponding feature points
- Mark relevant lines (one instance)
- Then: Learn PCA model of *relevant* graph parts

Usage

- Noisy, cluttered feature graphs
- Focus on „interesting“ subset
- Instance retrieval is still automatic

Overview

Introduction

Related Work

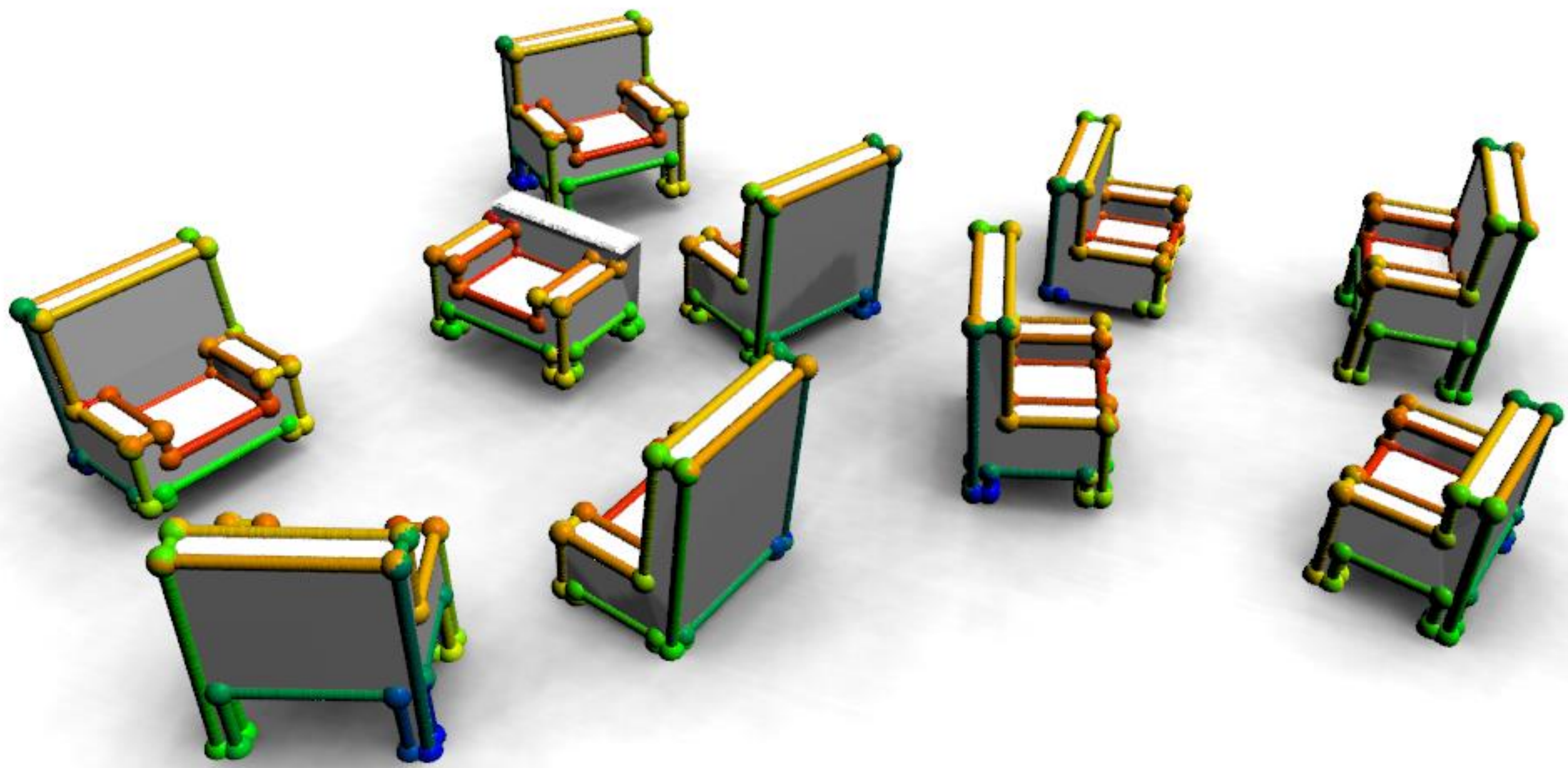
Subspace Symmetries

Detection Algorithm

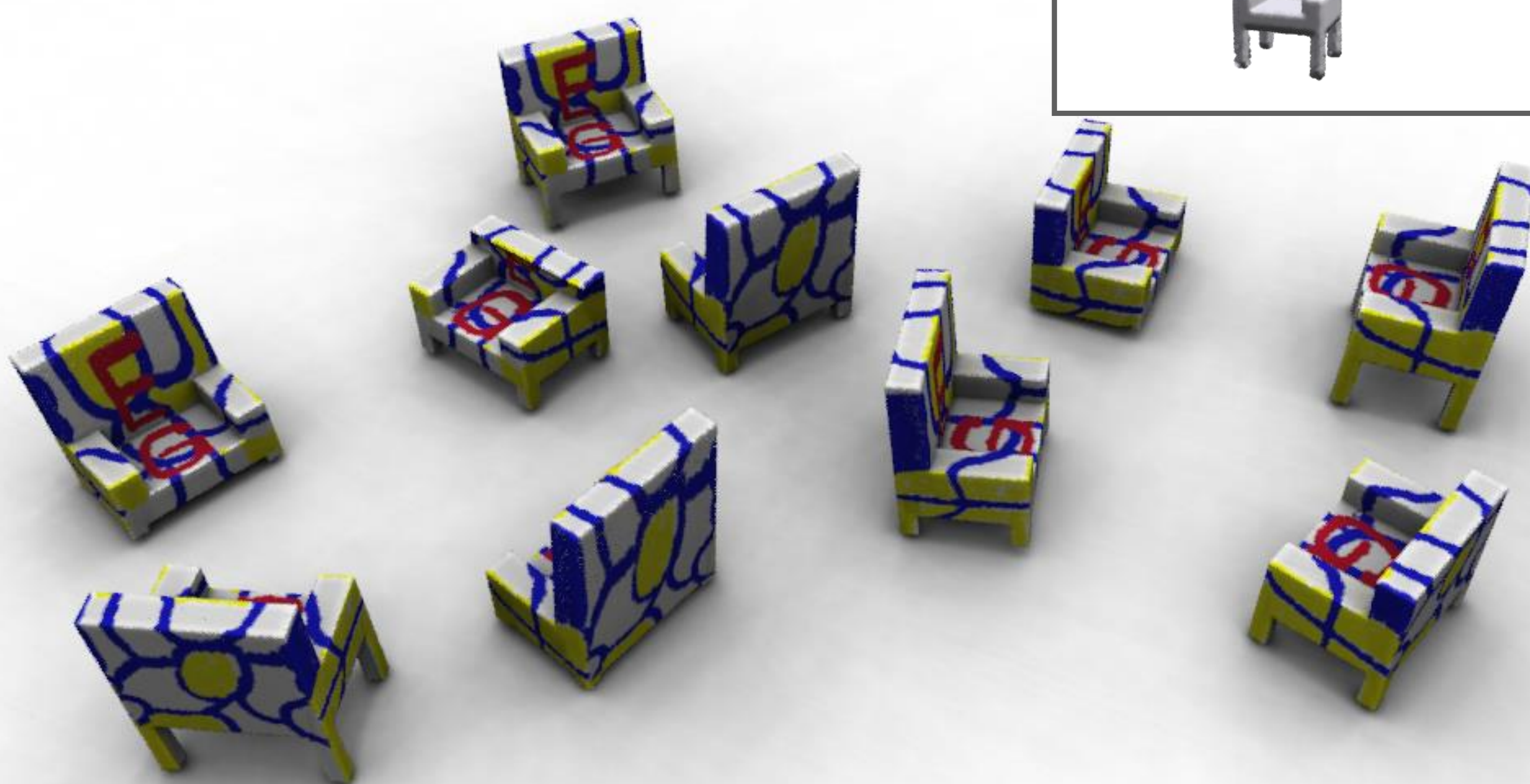
Results

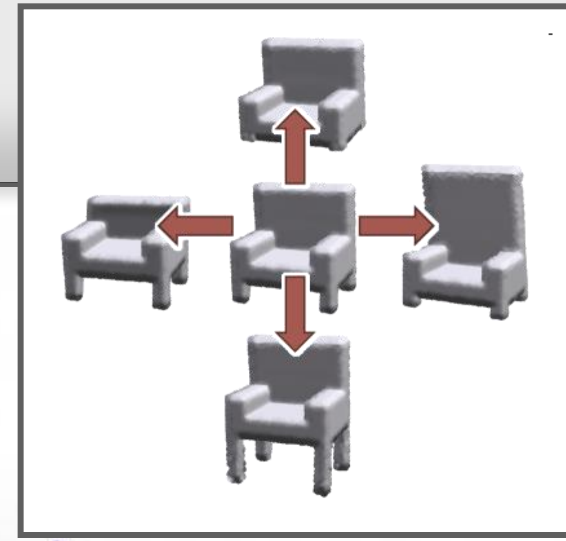
Conclusions

Chairs (synthetic)



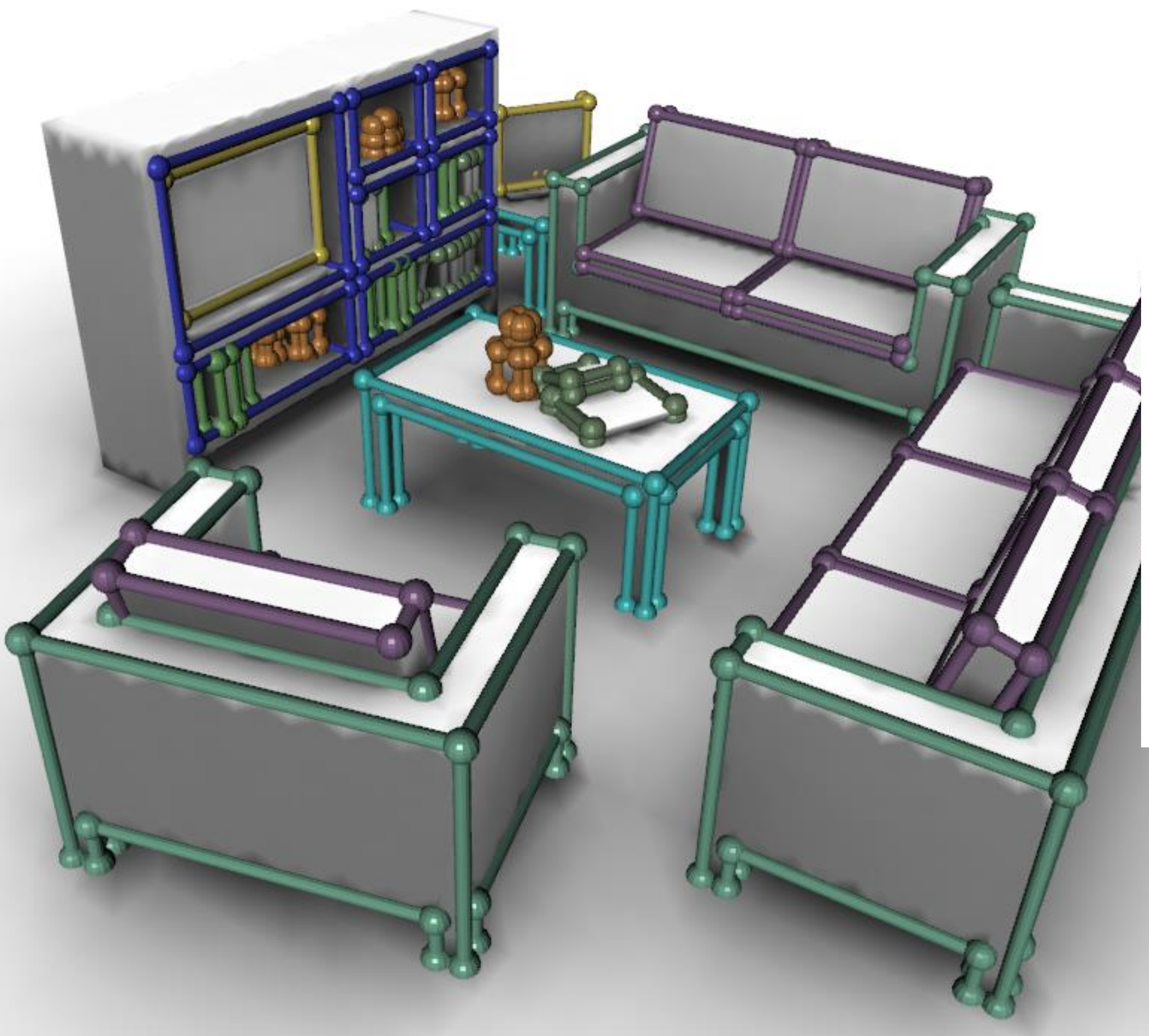
Chairs (synthetic)

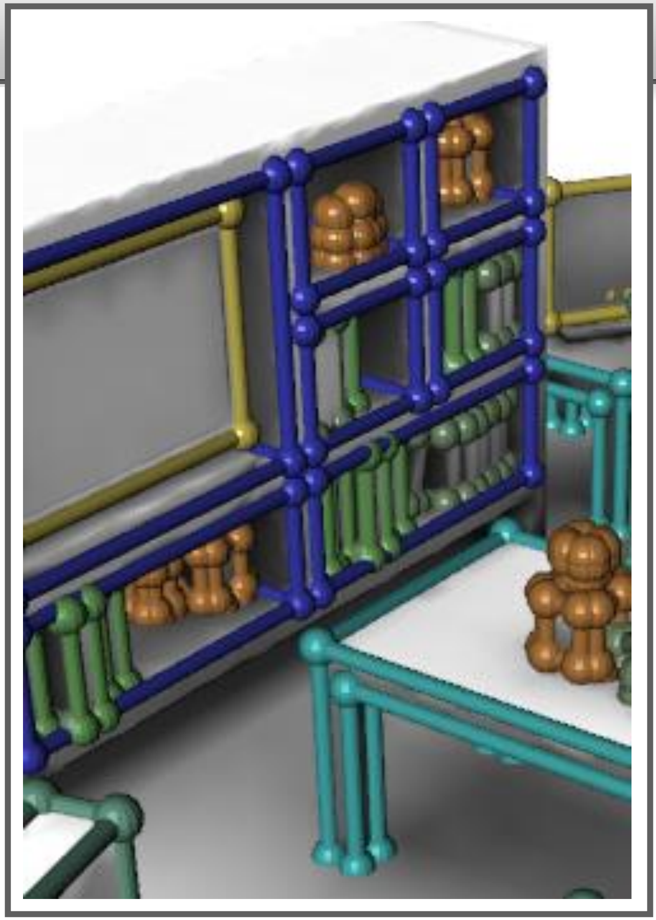




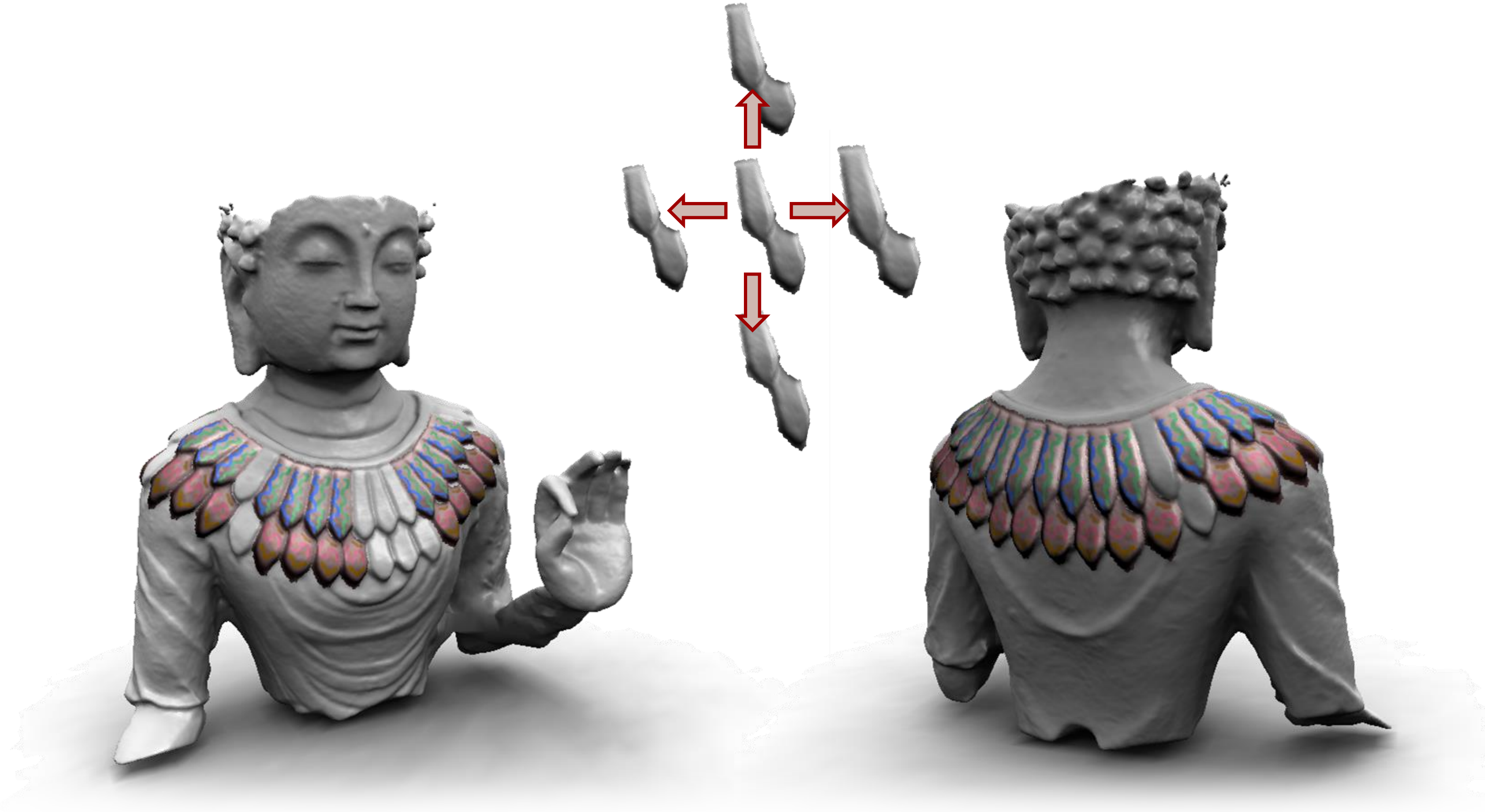
Living Room (synthetic)

Living Room (synthetic)

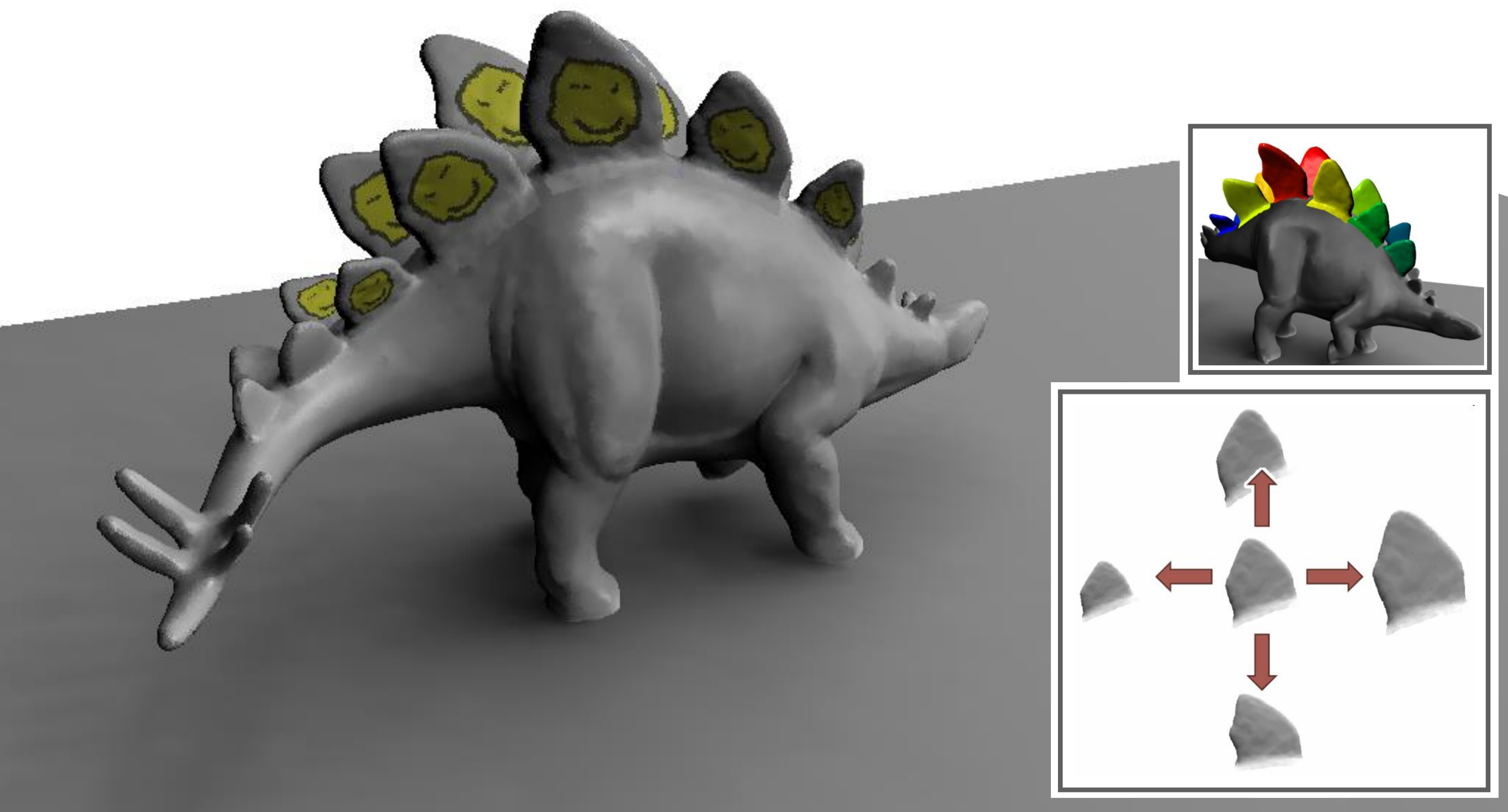




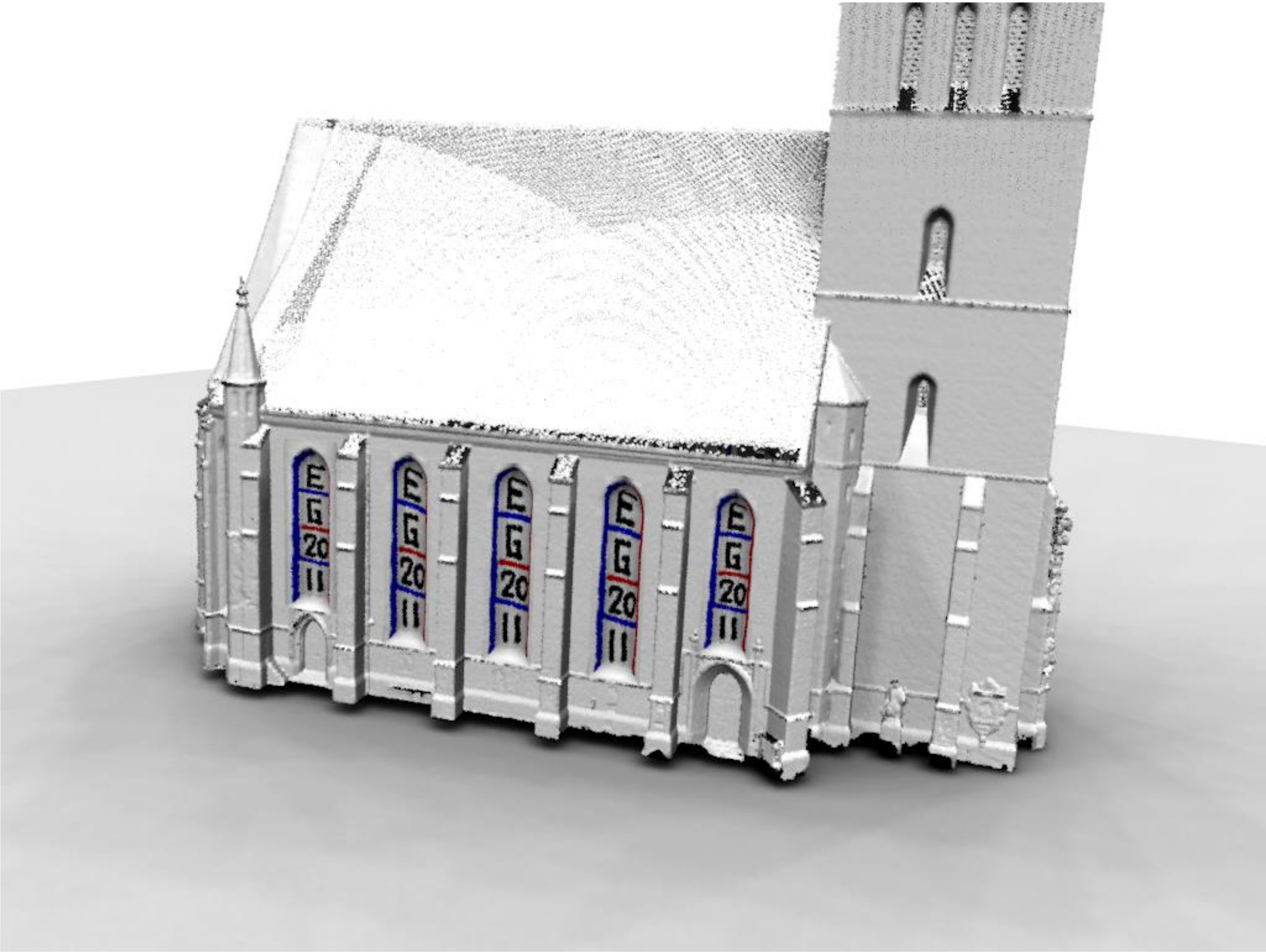
Statue (3D Scan)



Dino (3D Scan, Manual Training)



Church (3D Scan)



Overview

Introduction

Related Work

Subspace Symmetries

Detection Algorithm

Results

Conclusions

Conclusions

General notion of symmetry

- Important problem
- Proposal: subspace model

Heuristic graph matching algorithm

- Can get good results on clean input
  - Meshes and range data
  - Parameter dependent
- Training improves performance on ambiguous data

Future challange

- Provably efficient and effective solution