A Graph-Based Approach to Symmetry Detection

Alexander Berner¹, Martin Bokeloh¹, Michael Wand²,³, Hans-Peter Seidel³ and Andreas Schilling¹

¹WSI/GRIS University of Tübingen  ²Saarland University  ³Max-Panck-Institut Informatik
Overview

Talk Overview

• Introduction
• Symmetry Detection
• Results
• Conclusions & Future Work
Introduction
Problem Statement

Goal of this work: Symmetry Detection

• Detect similar parts in 3D geometry
• Decompose into building blocks

Many applications:

• Compression
• “Intelligent” modeling
• Reconstruction
Related Work

Transformation Voting

- [Mitra et al. 2006], [Podolak et al. 2006], [Loy et al. 2006], [Mitra et al. 2007], [Pauly et al. 2008]

Advantages & Disadvantages

- Good recognition performance
- Limitations:
  - Transformation space dimension grows with generality
  - Cluttering of transformation space for large models
Related Work

Divide-and-conquer approaches
  • [Simari et al. 2006], [Mitra et al. 2006]

Advantages & Disadvantages
  • Reduce complexity (cluttering)
  • Hierarchical output
  • Needs nested model structure to work
Related Work

Other approaches

• [Hubo et al. 2007], [Thrun et al. 2005], [Martinet et al. 2006]

• …
Related Work

Graph-based approaches

• [Felzenszwalb et al. 2005]
  ▪ 2D pictorial structures (graph of picture elements)
  ▪ Use for object recognition / tracking

• [Schnabel et al. 2008]
  ▪ Fit geometric primitives (sphere, cylinder...) to point cloud
  ▪ Graph of primitives
  ▪ Use for object recognition

Our approach: Graph-based symmetry detection
Symmetry Detection
Our Approach

Find symmetries by feature graphs

- Features detection
- Build feature graph
- Subgraph matching
  - Efficient randomized algorithm
- Geometry validation
  - Region growing
Feature Detection
Feature detection: Locally unique keypoints

- We need “many” stable keypoints
- “Slippage features” [Bokeloh et al. 2008]
  - Define keypoints by maximum stability of auto-alignment of local neighborhoods
- Feature descriptors: curvature histograms
Feature Graphs
Graph Generation

Feature Graph: k-nearest neighbors
Feature Graph Matching
Sub-graph matching:

- General problem: subgraph isomorphism
  - No efficient solution in general

In our case:

- Geometric similarity: spatial coherence
- We can find patterns with high probability by random sampling
- RANSAC-like algorithm
Overview

RANSAC matching:

• Choose a \textit{start edge} randomly (importance sampling)

• Find \textit{similar edges} using length and descriptors

• Walk on the feature graph from the \textit{start edge} and simultaneously from the \textit{similar edges} to validate symmetry
Sub-Graph Matching

Example scene with detected features
Sub-Graph Matching

Randomly chosen start edge
Sub-Graph Matching

From all other edges choose similar length edges with similar feature points
Sub-Graph Matching

Every edge transformation is known
Sub-Graph Matching

Expand adjacent edges
Sub-Graph Matching

Transform adjacent edges with given transformation
Sub-Graph Matching

Validate edges
Sub-Graph Matching

Expand further
Sub-Graph Matching

Transform and validate
Sub-Graph Matching

Final Result:

- Symmetric feature constellations
- Transformations
Noisy, incomplete data:

- Additional checks are necessary
- We add an ICP step
ICP Check in Point Clouds

Noisy, incomplete data:

- Additional checks are necessary
- We add an ICP step
ICP Check in Point Clouds

Noisy, incomplete data:

• Additional checks are necessary
• We add an ICP step ⇒ improved precision
Point Cloud Graph Result
Region Growing
From graphs to geometry:

• We have: symmetric feature constellations
• Check if geometry is symmetric, too.
  (Feature symmetry might be coincidental.)
Geometry validation:

• Region growing over the points
• Stop if:
  ▪ Residual distance exceeds threshold
  ▪ Regions collide
Results
Point Cloud Results

Clay house model after growing points
Point Cloud Results

Clay house model difference error
Point Cloud Results

Synthetic dataset with +/-10% Gaussian noise
Point Cloud Results

Historical artefact
Generalizations
2D Image Results

Bitmap image: *circuit diagram*

- Harris corners features
- Color histogram descriptors
- Same matching algorithm (same code) as before
2D Image Results

Bitmap image: “Ode to Joy”

 Manga text:

歡喜に寄せて
おお友よ、このような音ではない！
我々はもっと心地よい
もっと歡喜に満ち溢れる歌を歌おうではないか

Manga text:
Mesh Results

Triangle mesh: Power plant model

- Every triangle is a feature (key point)
- Side lengths as descriptor (sorted)
- Same matching algorithm (same code) as before
Triangle mesh: Power plant model
Mesh Results

Triangle mesh: Power plant model
Conclusions & Future Work
Conclusion

Conclusions

• A novel symmetry detection algorithm
• Based on feature graphs
• Randomized subgraph matching

Algorithm properties

• Feature-based
  ▪ General modalities (images, triangles, points)
  ▪ Low memory footprint
• Matching can be generalized easily
Future Work

More general matching
- Isometric matching: check intrinsic distances
- Deformable matching
- Graph isomorphism: more in-class variation

Applications
- Using symmetries for reconstruction
- Apply to large datasets
歓喜に寄せ
おお友よ、このような音ではない！
我々はもっと心地よい
もっと歓喜に満ち溢れる歌を歌おうではないか

歓喜に寄せ
おお友よ、このような音ではない！
我々はもっと心地よい
もっと歓喜に満ち溢れる歌を歌おうではないか

Questions?