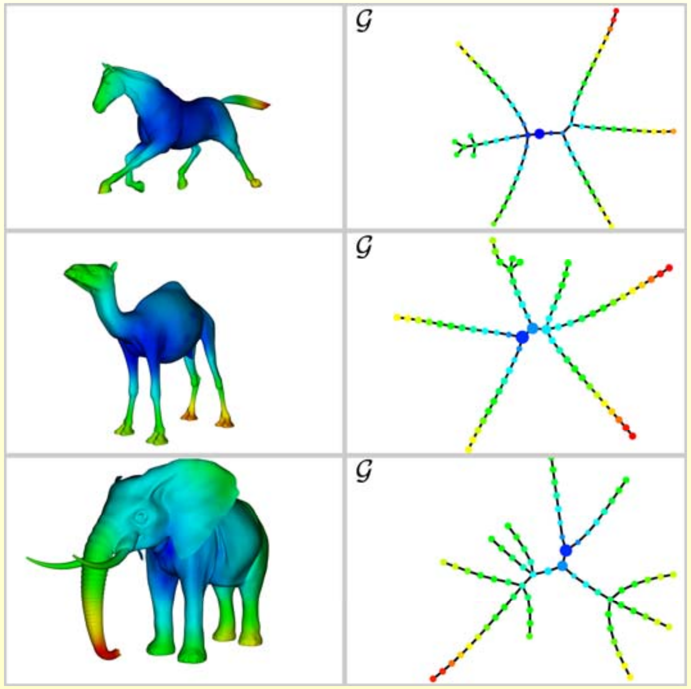
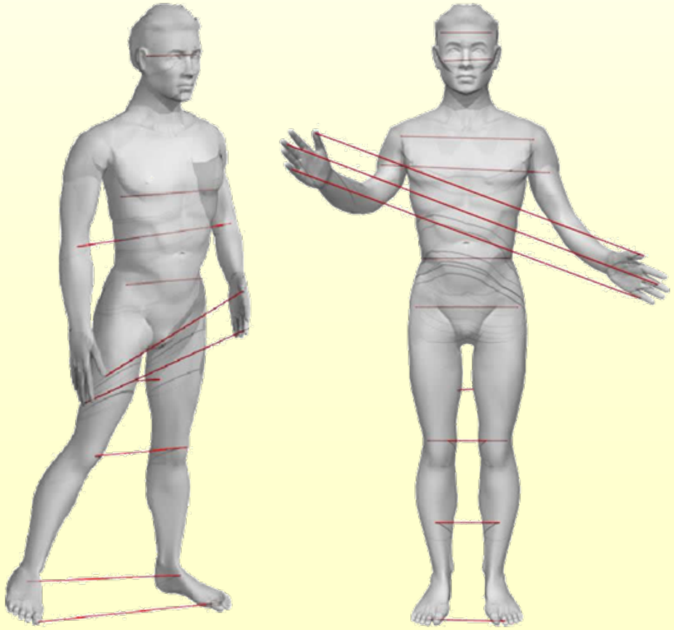
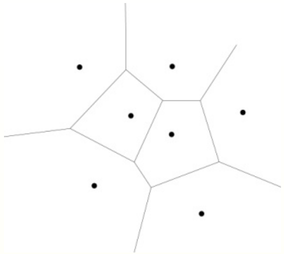
Global Structure Discovery in Sampled Spaces

NSF/DHS
FODAVA 2008

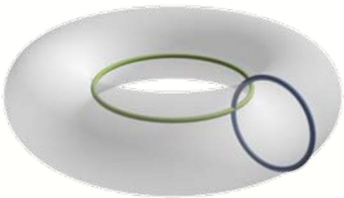

Leonidas Guibas
Gunnar Carlsson
Stanford University







Project Goals



- Bring tools from Computational Geometry and Topology to the analysis and visualization of massive, distributed data sets
- Perform global structure discovery on such data
  - Produce meaningful topological maps over the data
  - Extract structural self-similarities of the data (symmetries, repeated patterns)
- Exploit this discovered structure in enabling visual exploration and human interaction with the data

A Few Quick Vignettes from Current Work

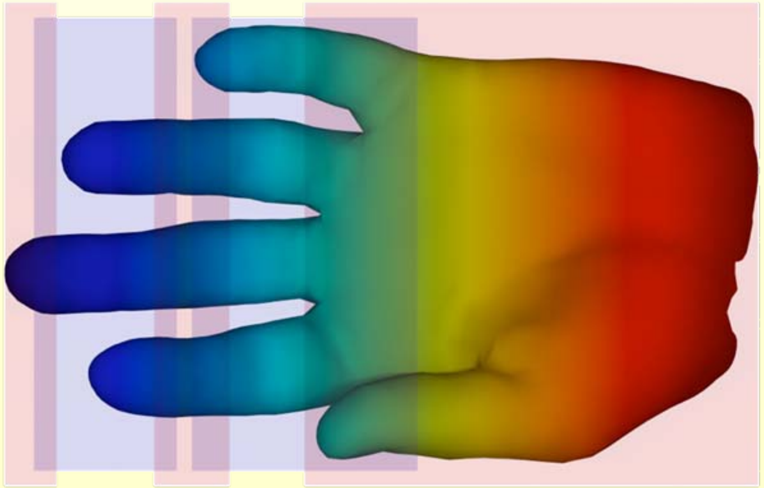
- I. Morse theory for combinatorial views of data
- II. Mining in transform spaces:
  - Partial and approximate symmetry extraction
  - Repeated pattern detection

- III. Scalar field analysis over metric spaces
- IV. Fingerprints for lightweight distributed data fusion

Mostly for 3D point clouds – but with a view towards high-d extensions

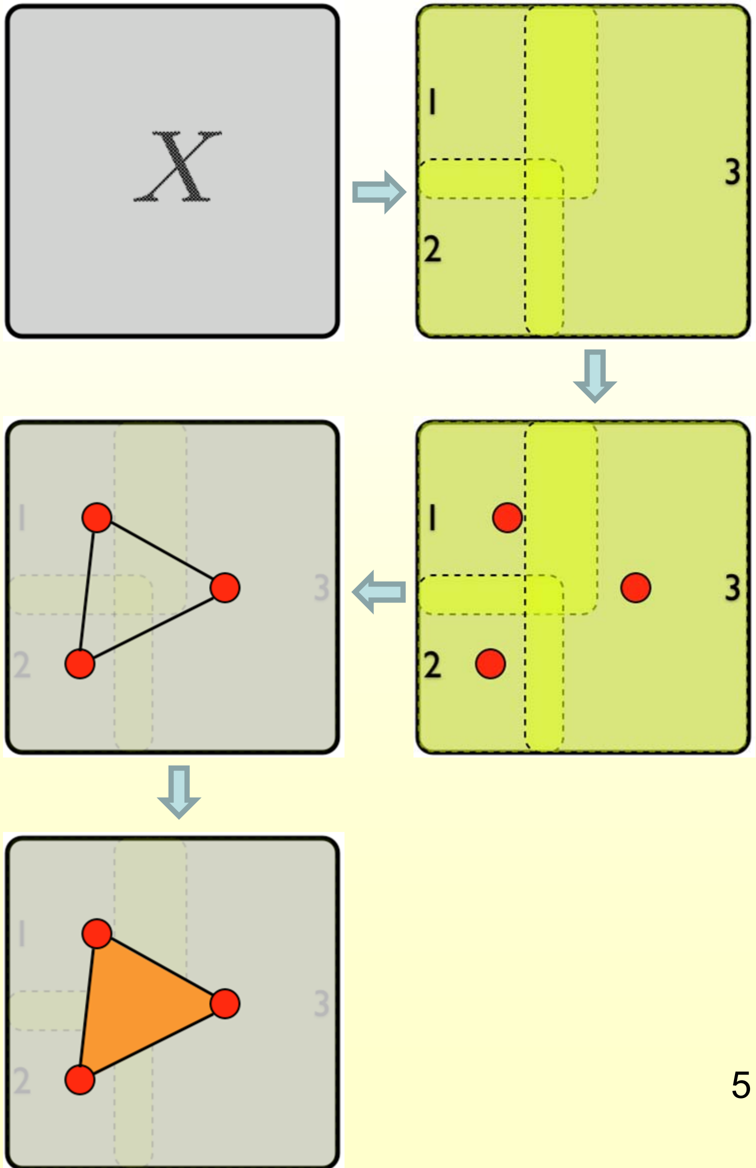
I. Mapper: Morse Theory for Combinatorial Views of Data

[G. Carlsson, F. Memoli, G. Singh]



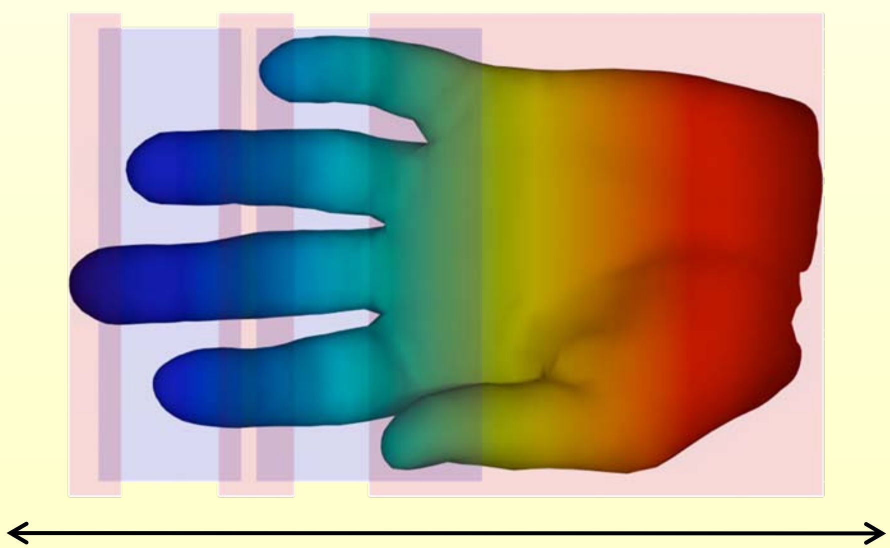
Simplicial Complexes

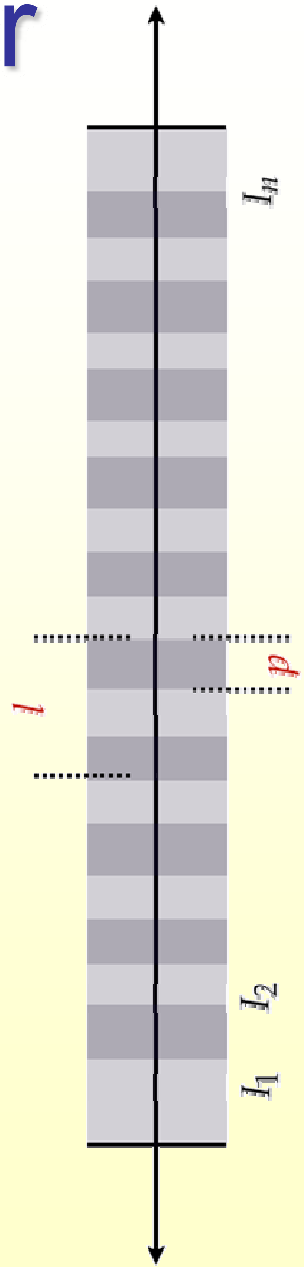
- We cover a space X with a system U of open sets
- We form a simplicial complex from the intersection patterns of these sets
- This is the nerve N of U, or the Čech complex of the set system
- Under some mild conditions, the topology of N captures that of X



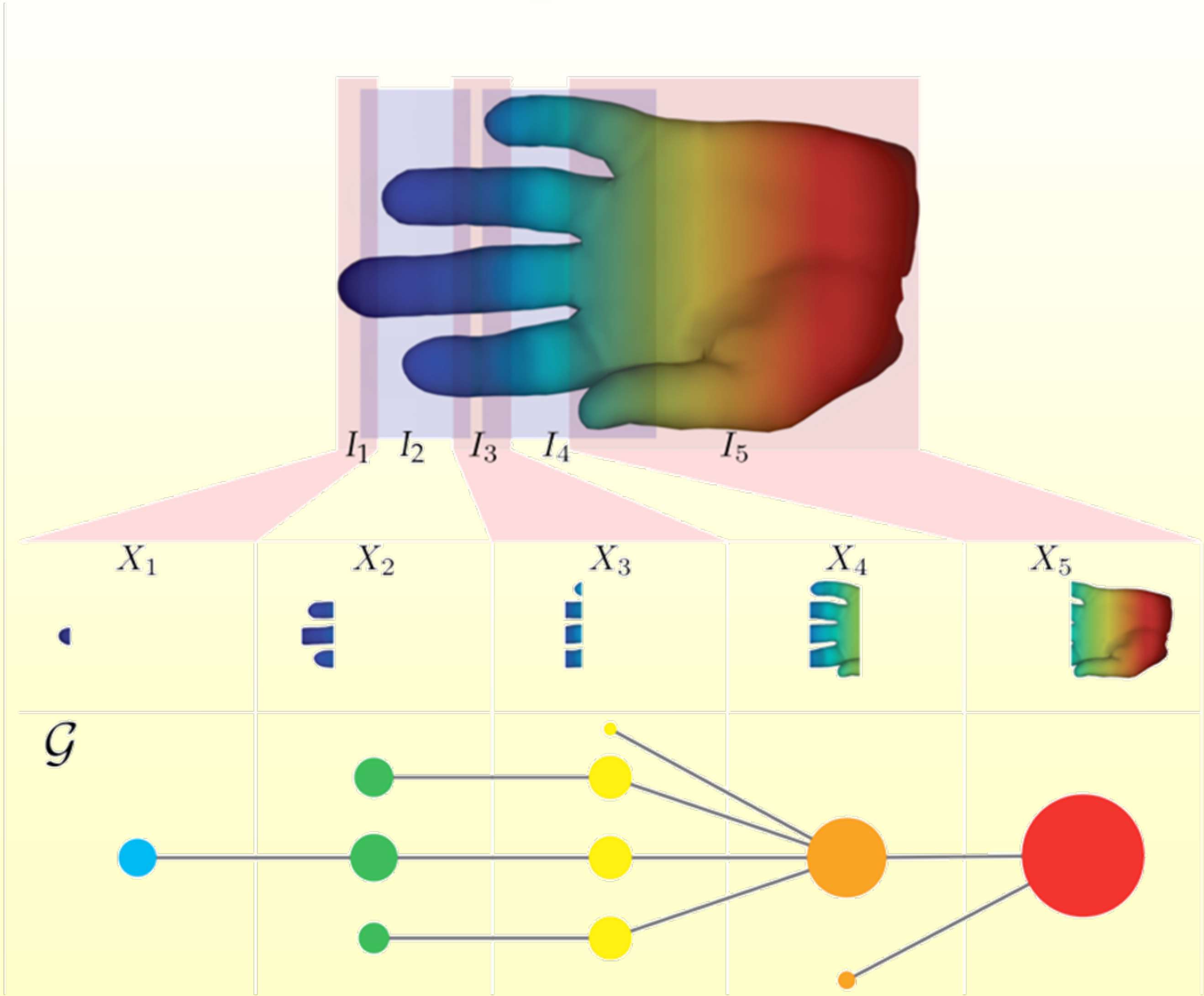
Open Covers from Filter Functions

- Consider a filter function $f : X \mapsto R$
- Cover R with intervals
- Use connected components of their inverse images for the X cover



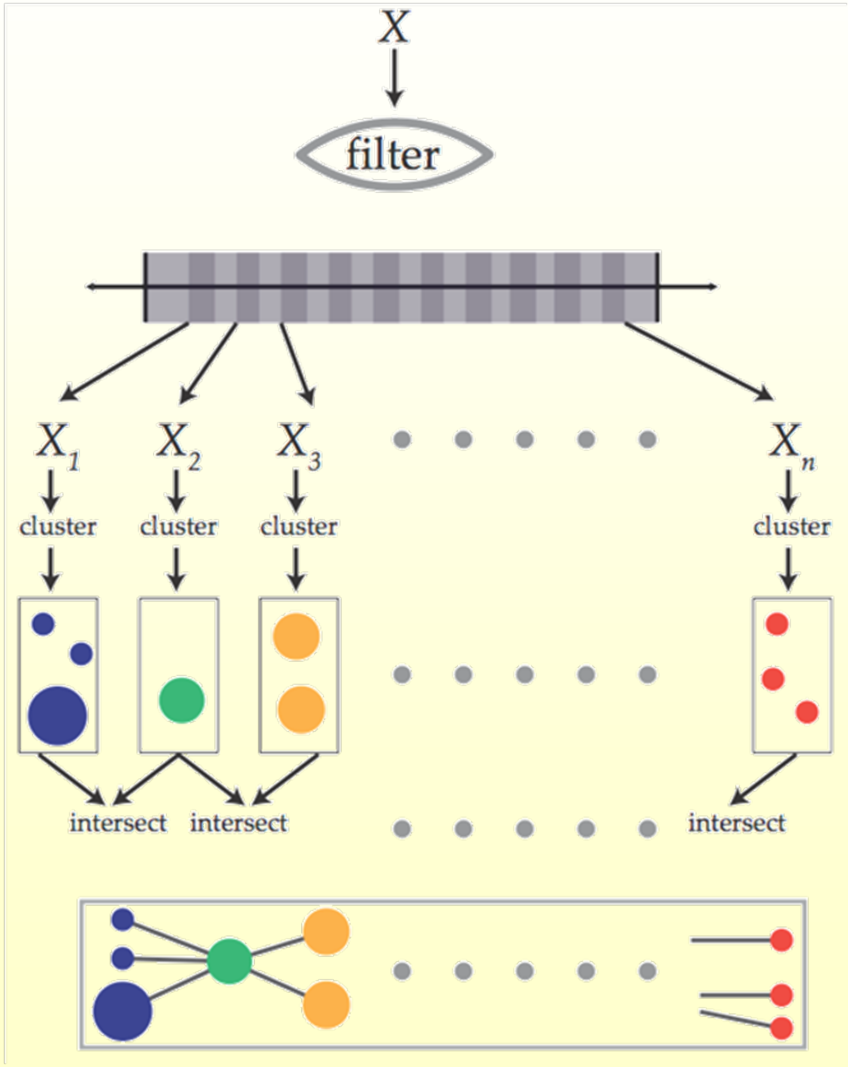


Overlap Structure of the Components



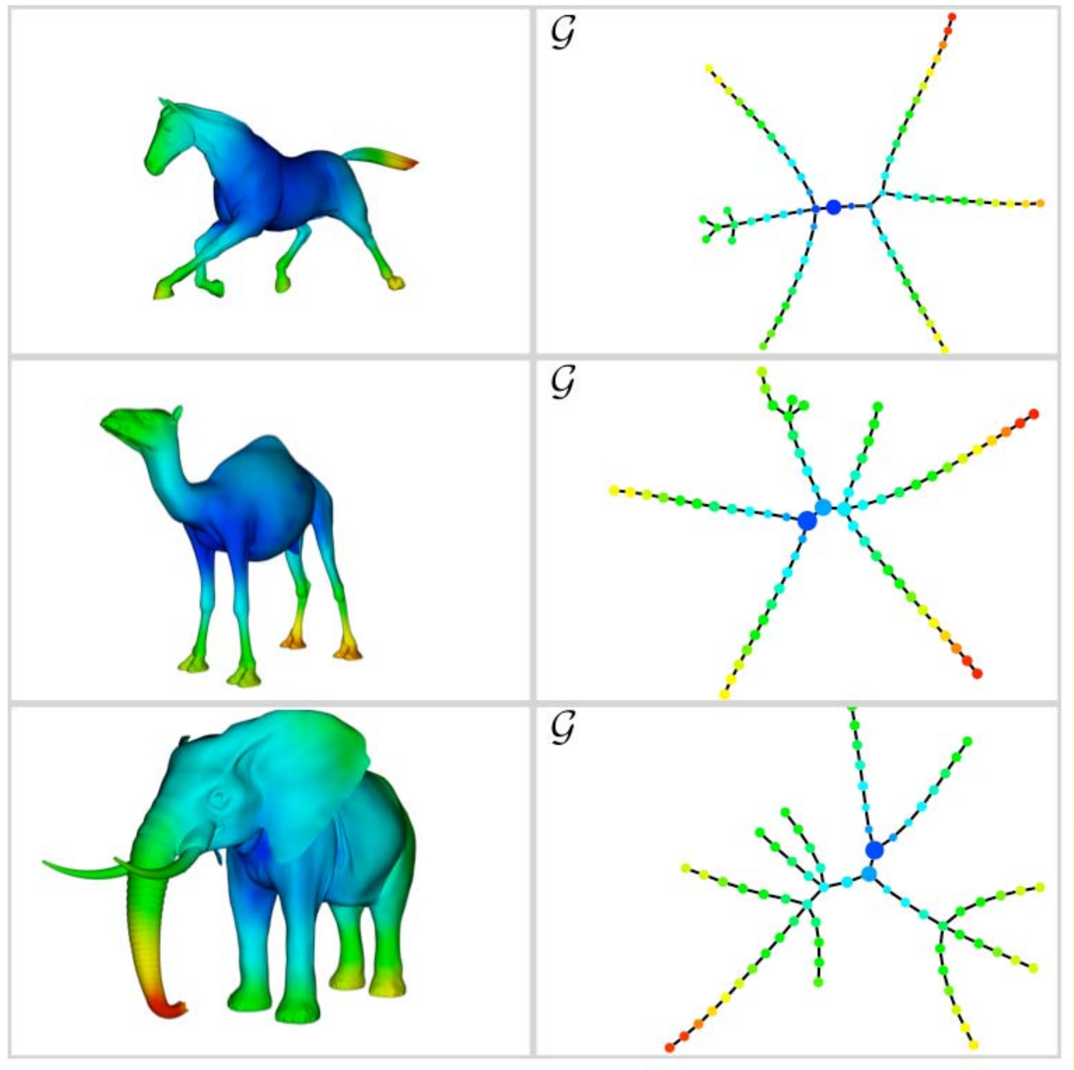
The Mapper Recipe

- Mapper
  - Combinatorial
  - Visual
  - Scalable

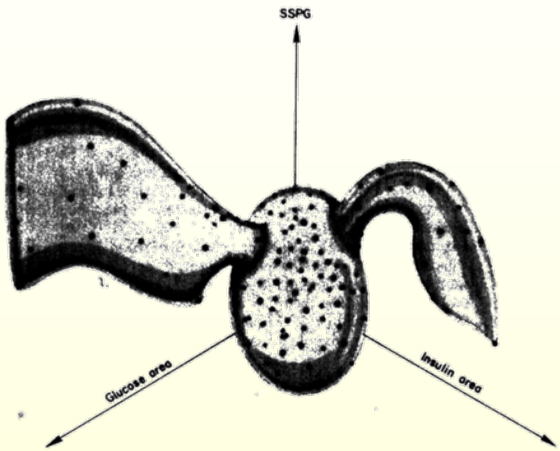


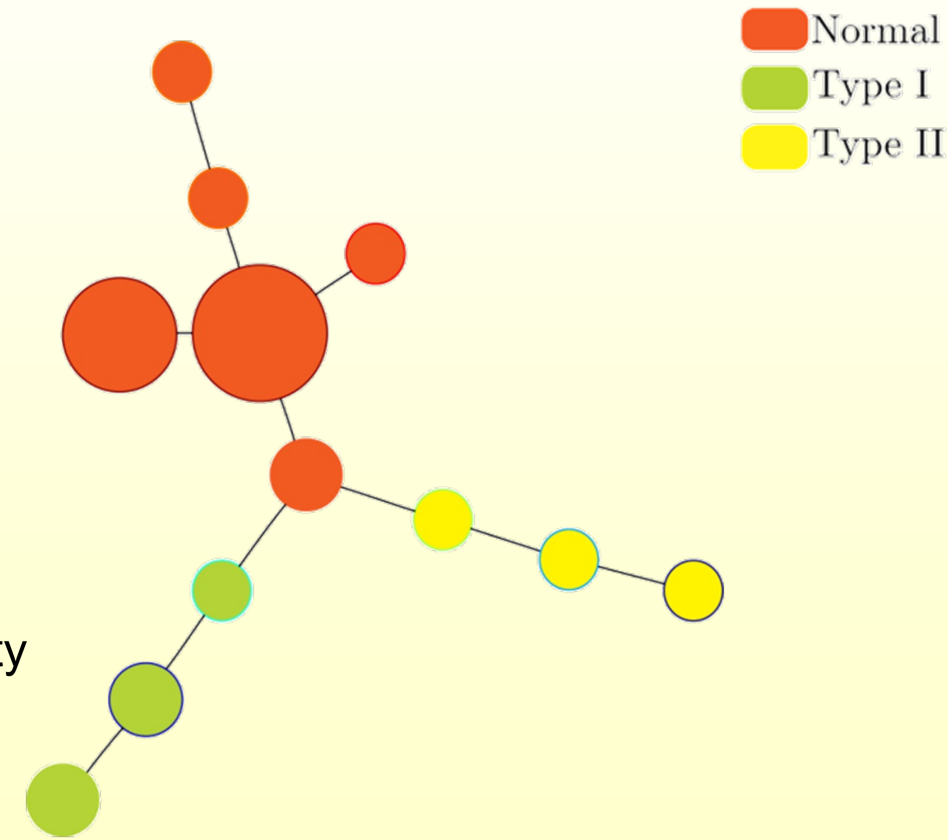
Clustering replaces connected components in sampled spaces

“Eccentricity” Filter Function



Miller-Reaven Diabetes Study





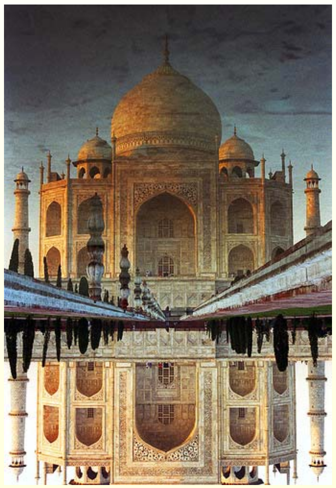
Mapper on the same data, using L^2 distance and a Gaussian density estimator as the filter function

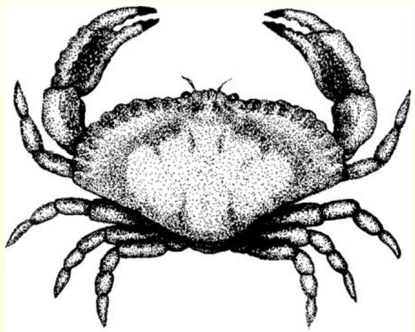
II. Mining in Transform Space

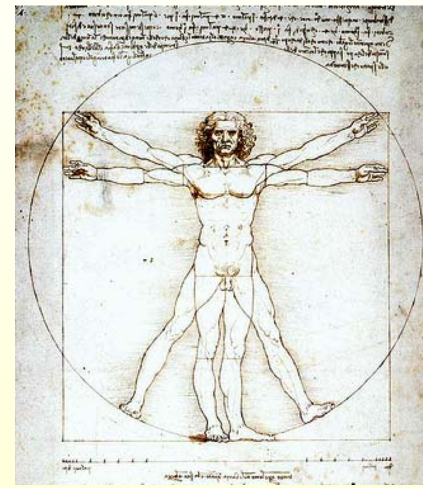
A. Partial and Approximate Symmetry Extraction

[N, Mitra, L. G., M. Pauly]

Symmetries and Regular Patterns In Natural and Man-Made Objects





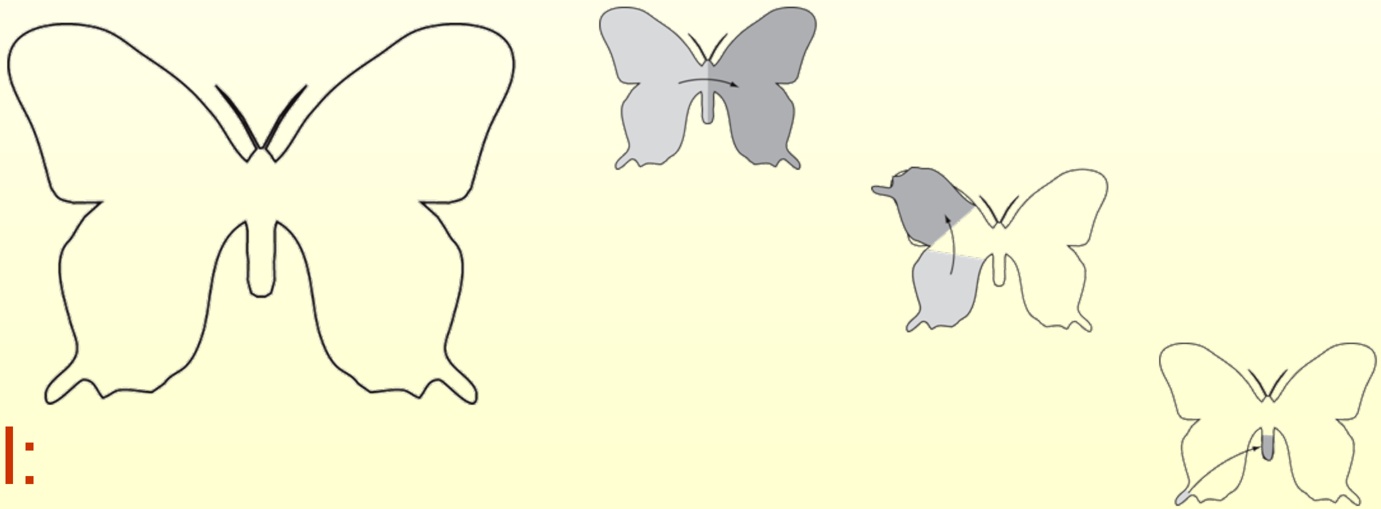


"Symmetry is a complexity-reducing concept [...]; seek it everywhere.
Alan J. Perlis

Partial/Approximate Symmetry Detection

Given:

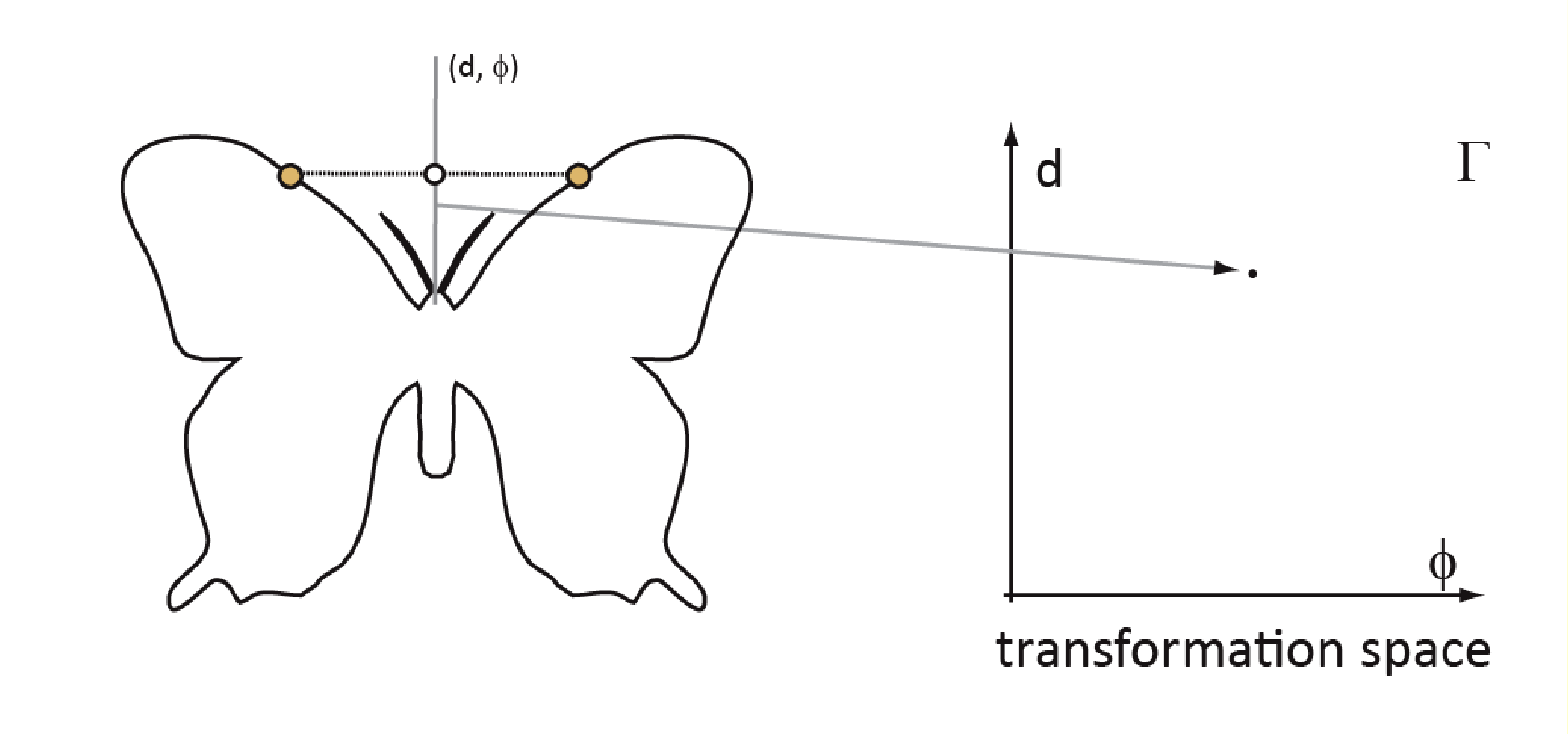
Object/shape (represented as point cloud, mesh, ...)



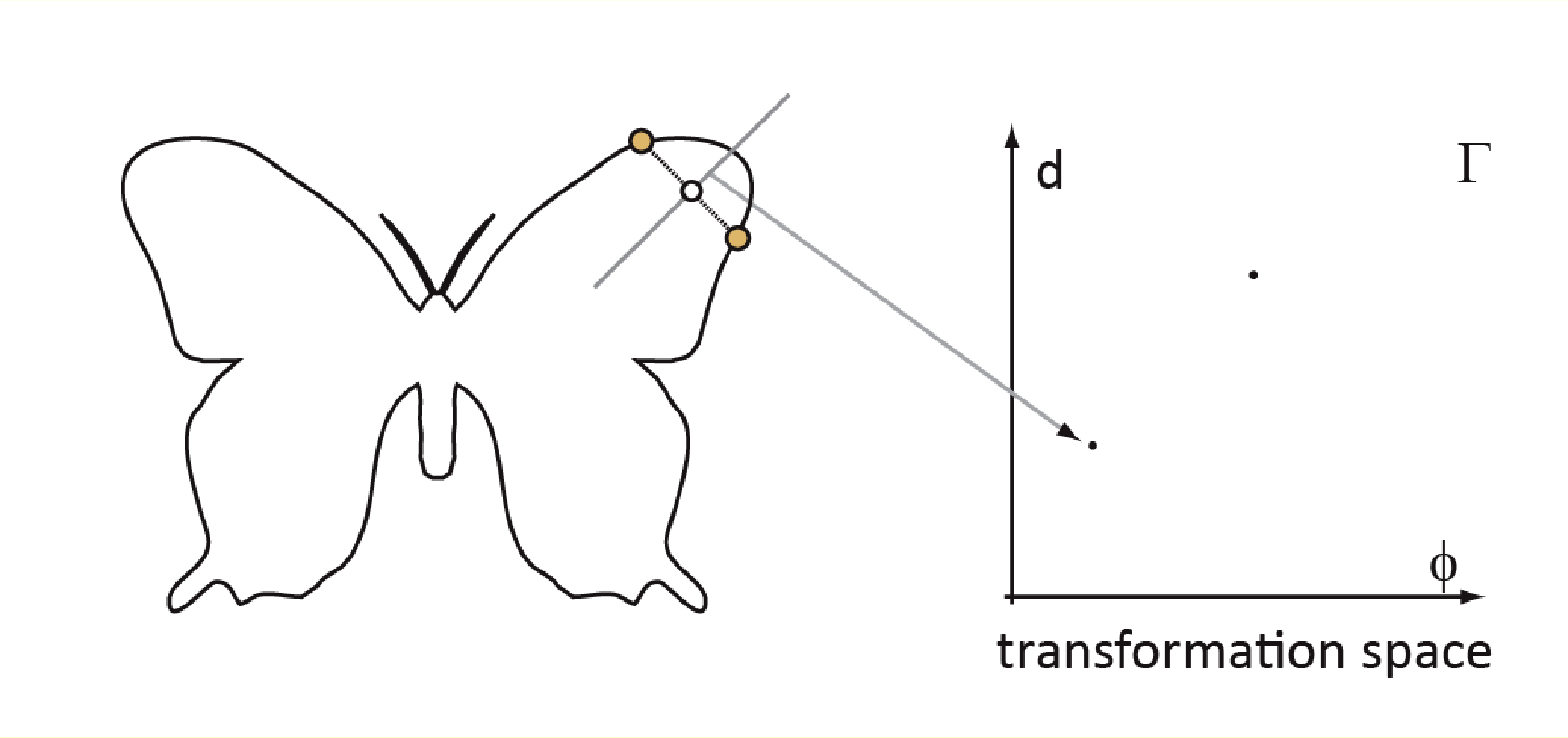
Goal:

Identify and extract similar (symmetric) patches of possibly different sizes, across different resolutions

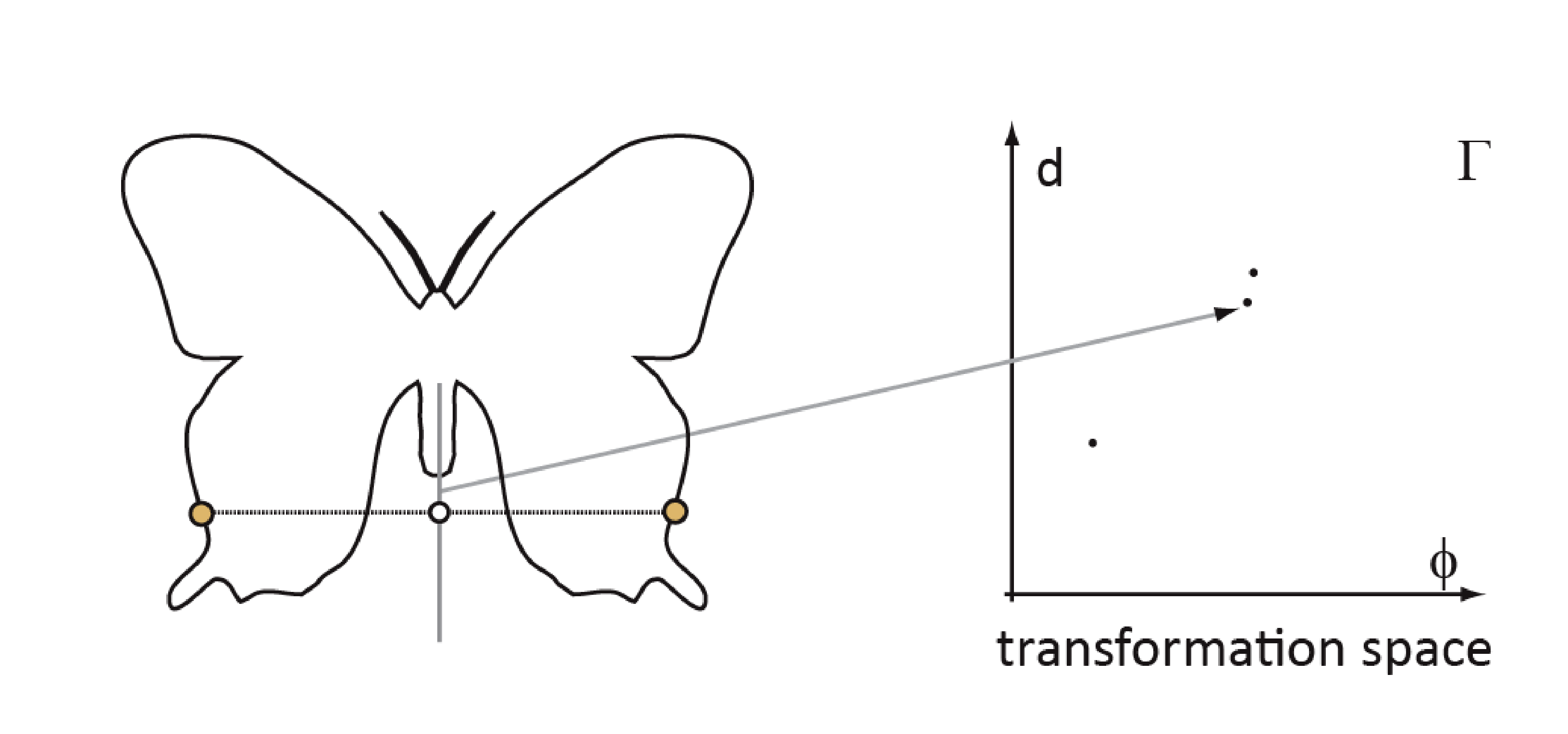
Transform Voting Example: Reflective Symmetry



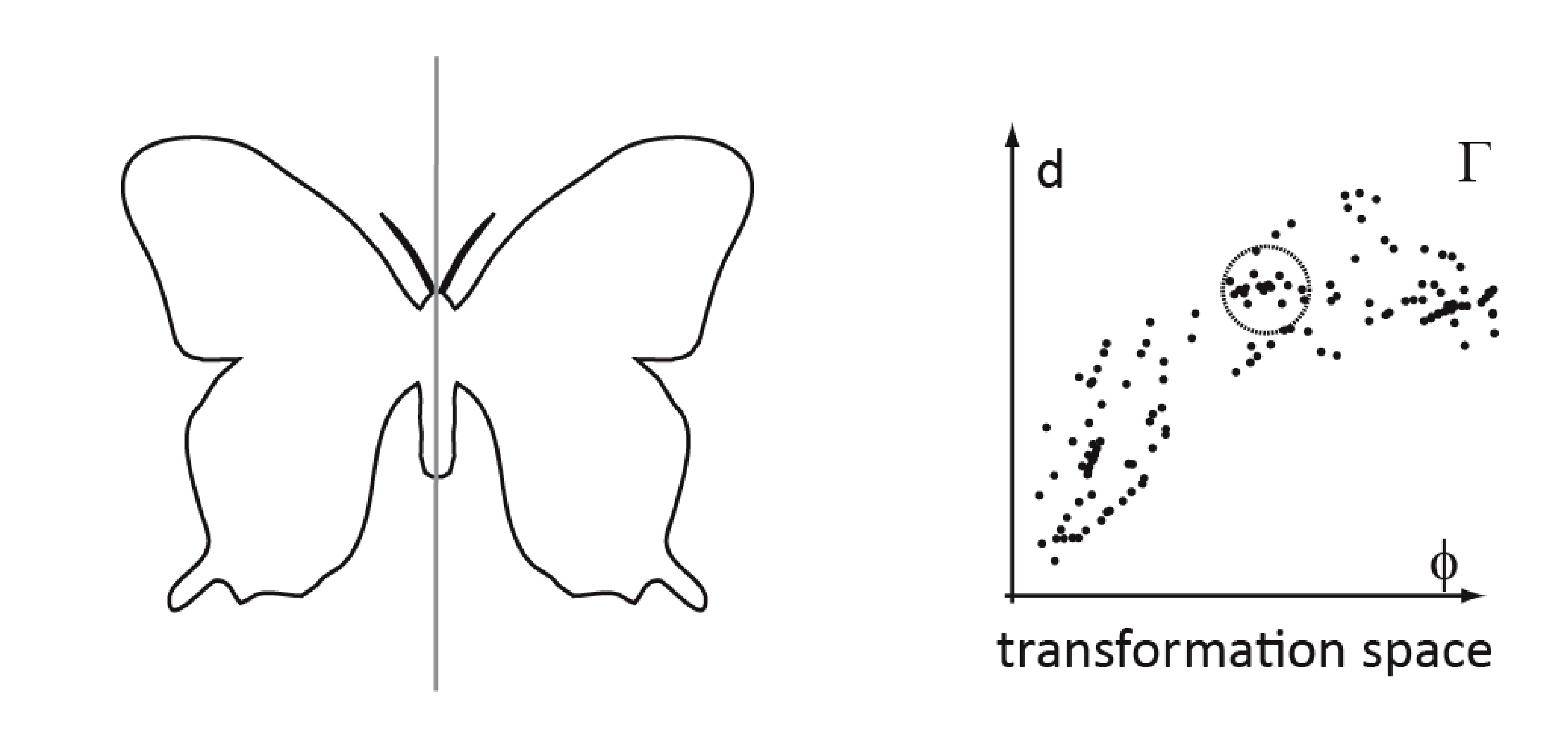
Reflective Symmetry : Voting Continues



Reflective Symmetry : Voting Continues

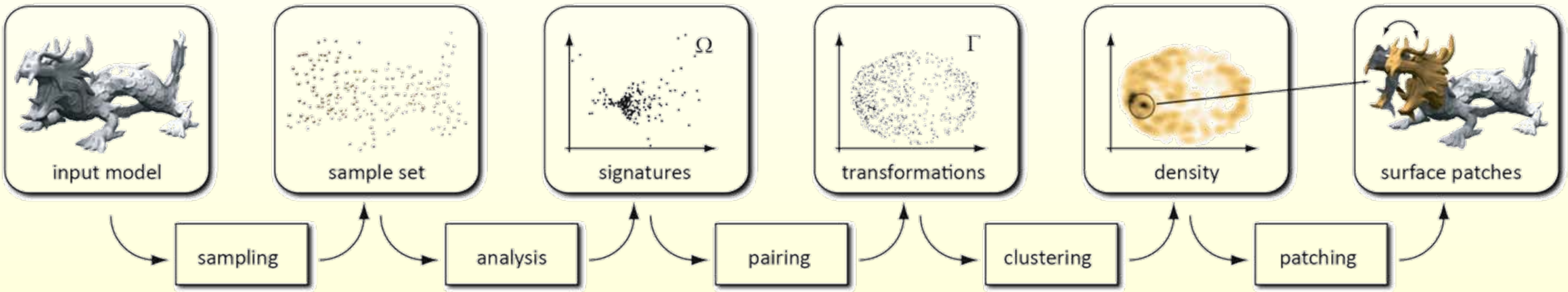


Reflective Symmetry : Largest Cluster



- Height of cluster → size of patch
- Spread of cluster → approximation level

Pipeline

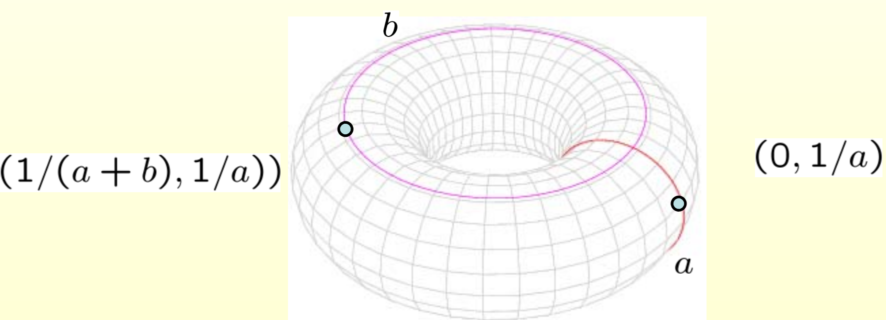


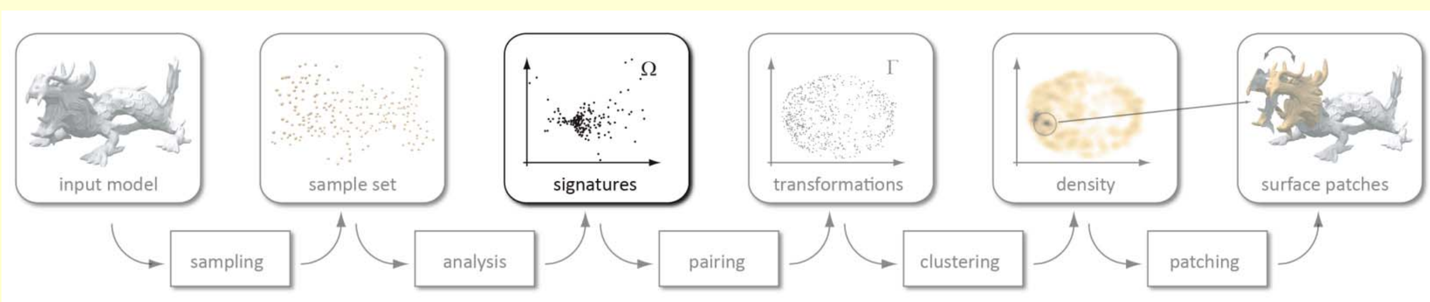
input model
sample set
signatures
Ω
transformations
Γ
density
surface patches
sampling
analysis
pairing
clustering
patching

Pruning: Local Signatures

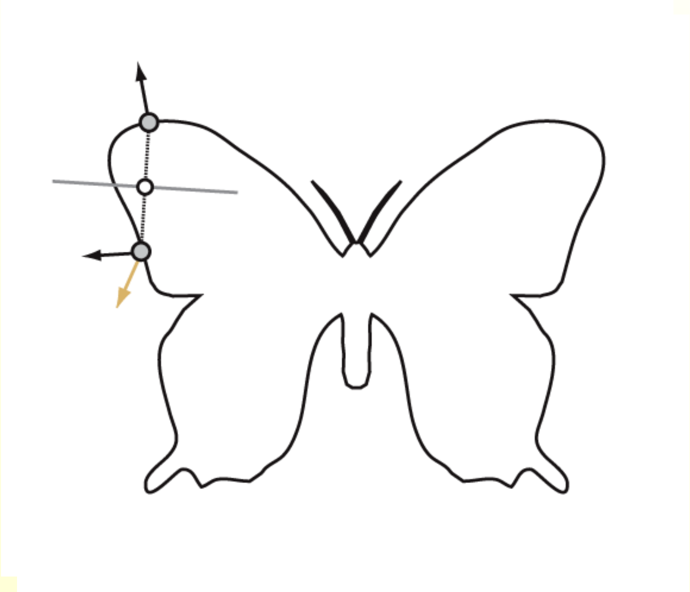
- Local signature → invariant under transforms
- Signatures disagree → points don't correspond

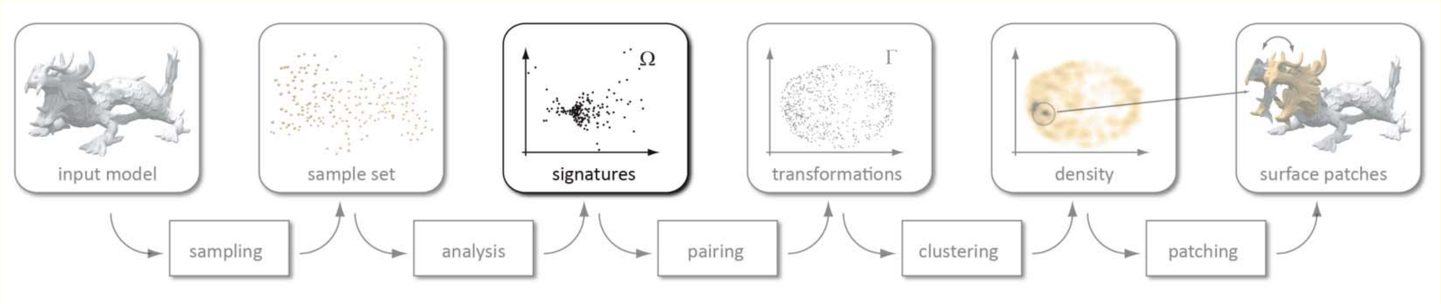
Example: use (κ_1, κ_2) for curvature based pruning



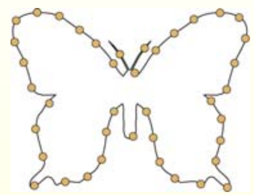


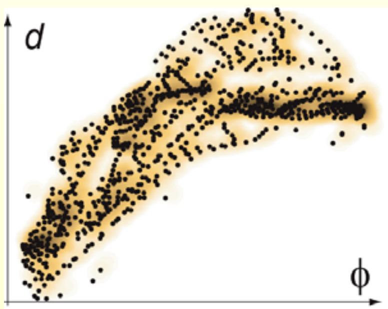
Reflection: Normal-Based Pruning



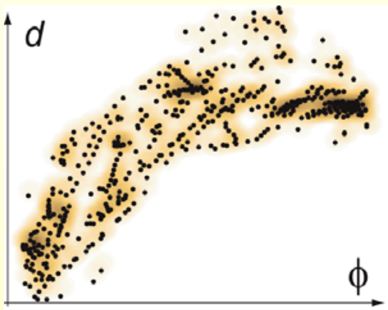


Point Pair Pruning

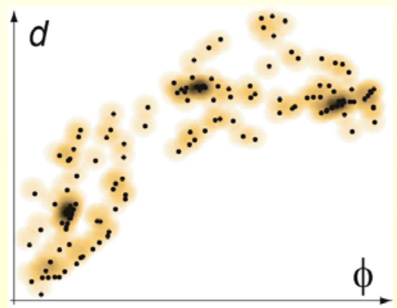




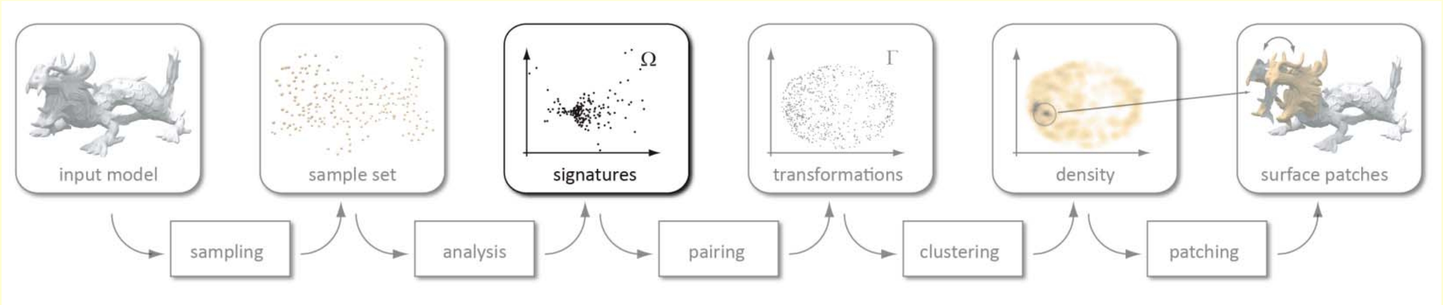
all pairs



curvature based



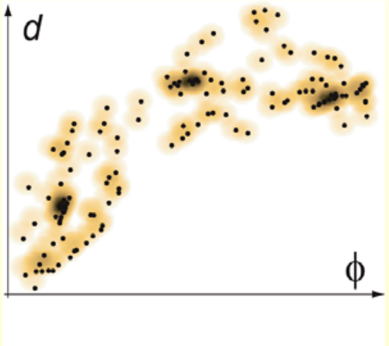
curvature + normal based

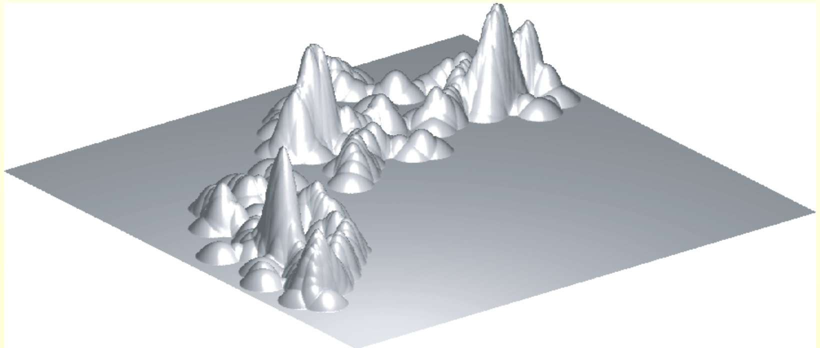


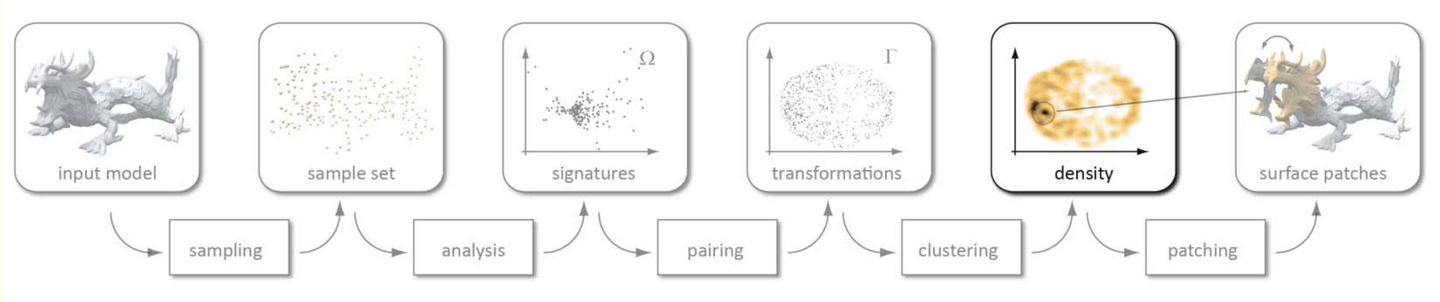
Mean-Shift Clustering

Kernel:

- Type → radially symmetric hat function
- Radius

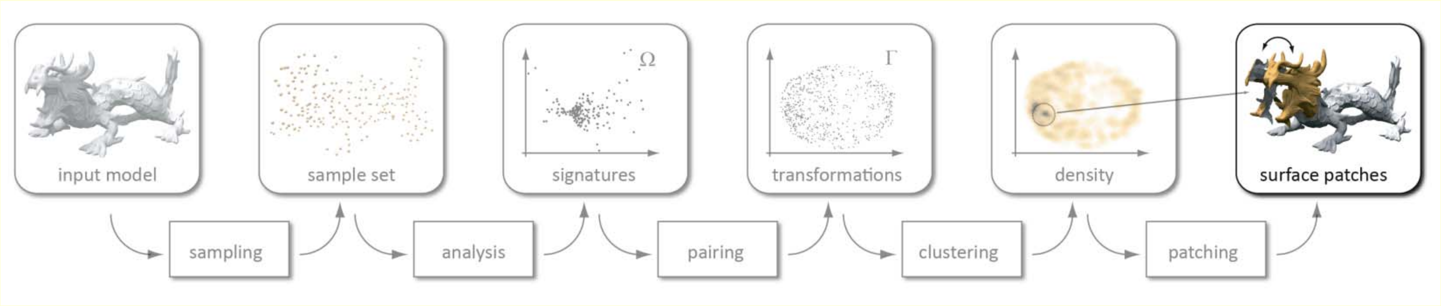






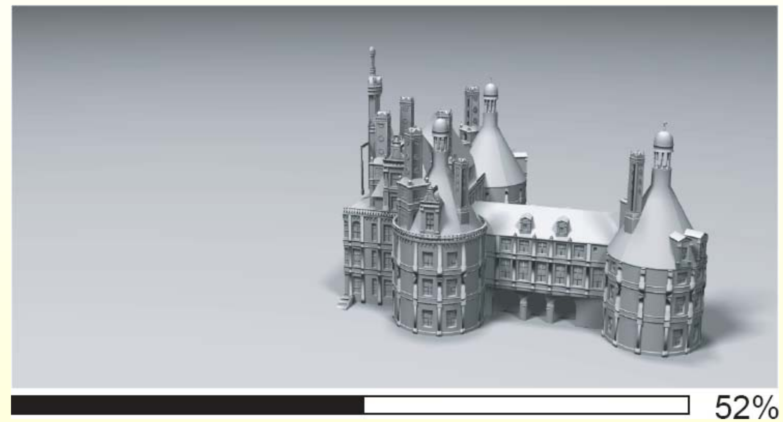
Verification

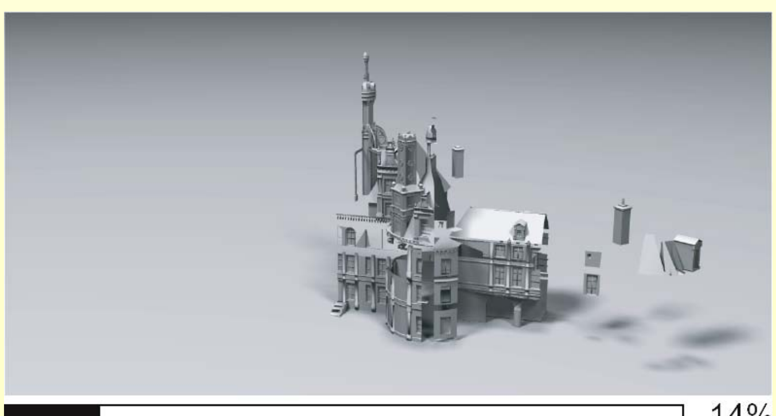
- Clustering gives a good guess of the dominant symmetries
- Suggested symmetries need to be verified against the data
- Locally refine transforms using ICP algorithm [Besl and McKay `92]

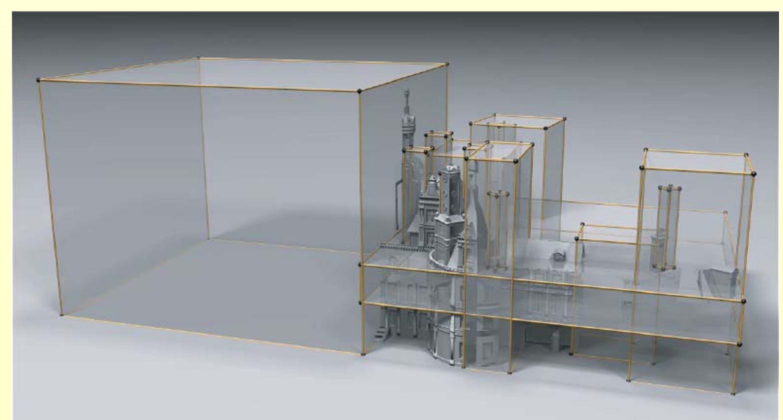


Compression: Chambord

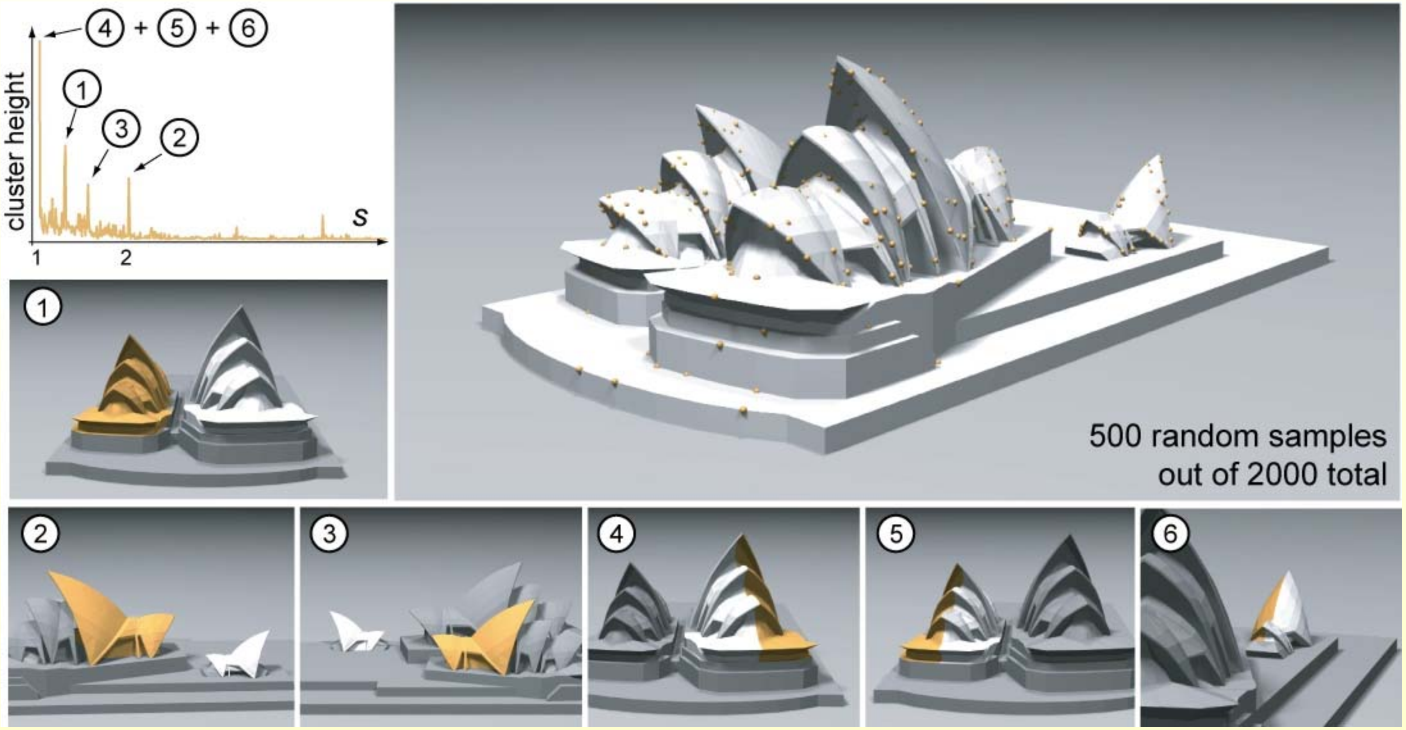
Compression: Chambord





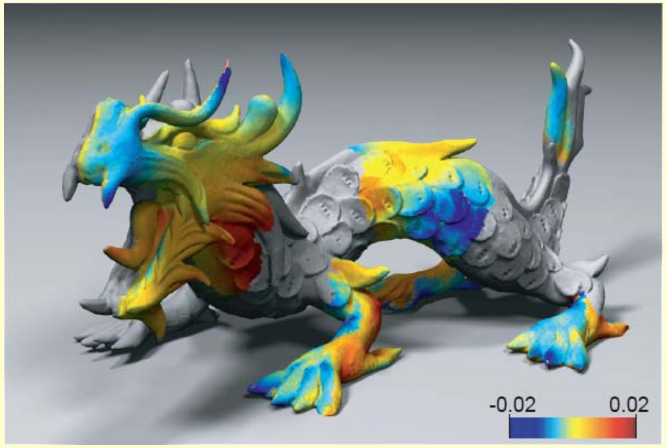


Opera


4 + 5 + 6
cluster height
1
3
2
s
1
2
1
500 random samples
out of 2000 total
2
3
4
5
6

Approximate Symmetry: Dragon

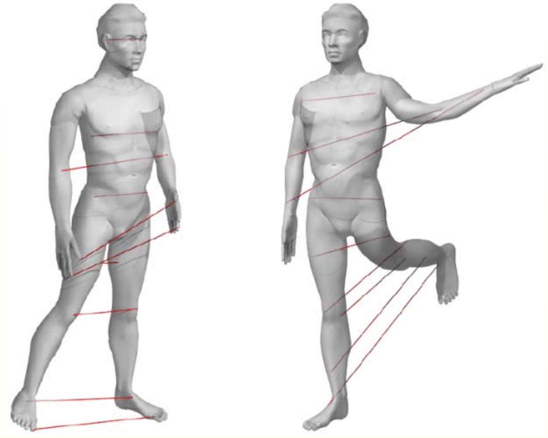
detected symmetries



correction field

Extrinsic vs. Intrinsic Symmetries

Extrinsic symmetry



Intrinsic symmetry

• Invariance under translation, rotation, reflection and scaling (Isometries of the ambient space)

• Break under isometric deformations of the shape

• Invariance of geodesic distances under self-mappings. For a homeomorphism $T : O \rightarrow O$

$$g(\mathbf{p},\mathbf{q}) = g(T(\mathbf{p}),T(\mathbf{q}))\ \forall\ \mathbf{p},\mathbf{q} \in O$$

• Persist under isometric deformations

• Introduced by Raviv et al. in NRTL 2007

[M. Ovsjanikov, J. Sun, L. G.]

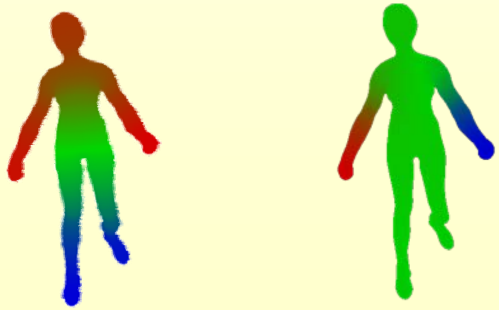
Global Intrinsic Symmetries

- Signature space
  - For each point **p** define its signature s(**p**) [Rustamov, SGP 2007]

$$s(\mathbf{p}) = \left(\frac{\phi_1(\mathbf{p})}{\sqrt{\lambda_1}}, \frac{\phi_2(\mathbf{p})}{\sqrt{\lambda_2}}, \ldots, \frac{\phi_i(\mathbf{p})}{\sqrt{\lambda_i}}, \ldots \right)$$

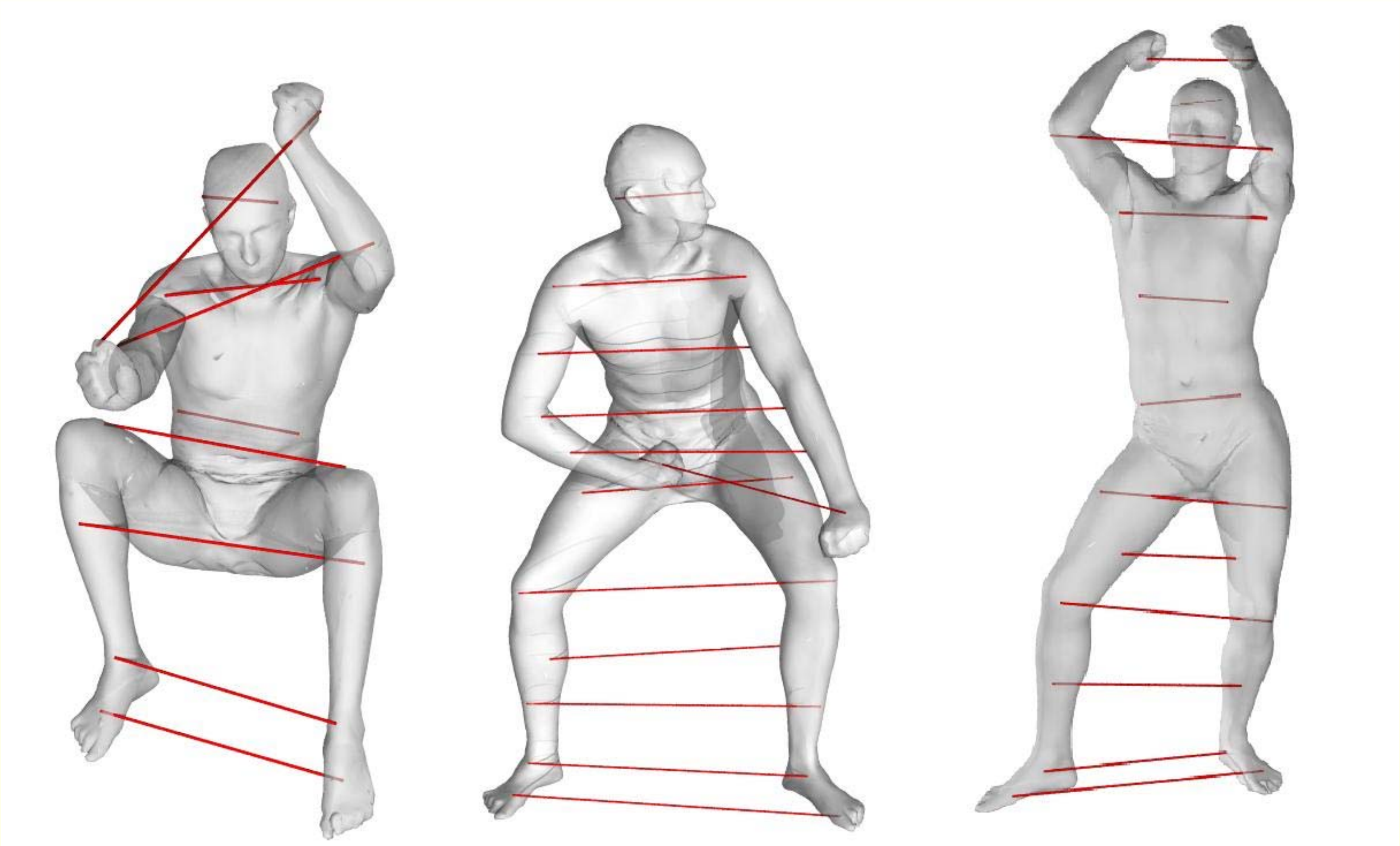
  - $\phi_i(\mathbf{p})$ is the value of the *i*-th eigenfunction of the Laplace-Beltrami operator at **p**
  - Invariant under isometric deformations
- Main Observation: Intrinsic symmetries of the object become extrinsic symmetries of the signature space.

1. $\phi = \phi \circ T$: **positive** eigenfunction
2. $\phi = -\phi \circ T$: **negative** eigenfunction
3. λ is a repeated eigenvalue



Positive Negative

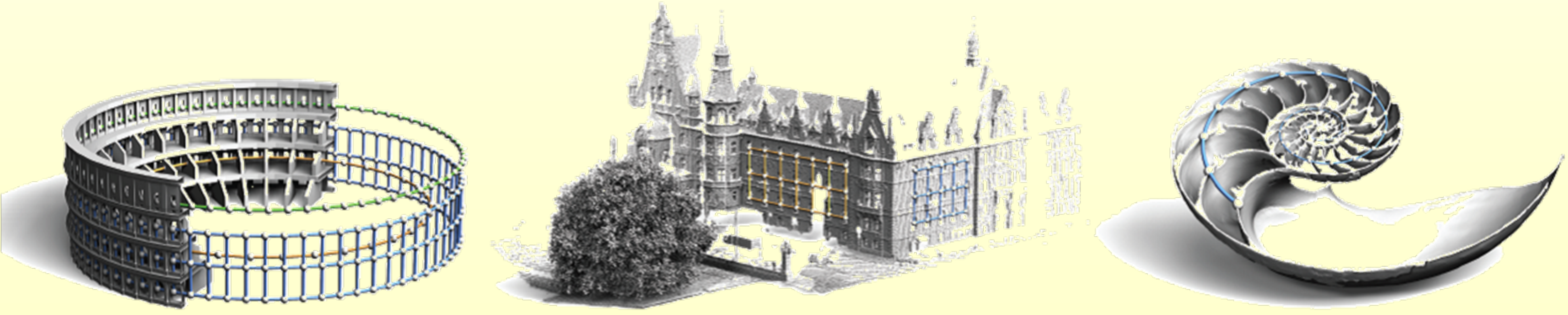
Global Intrinsic Symmetries



II. Mining in Transform Space

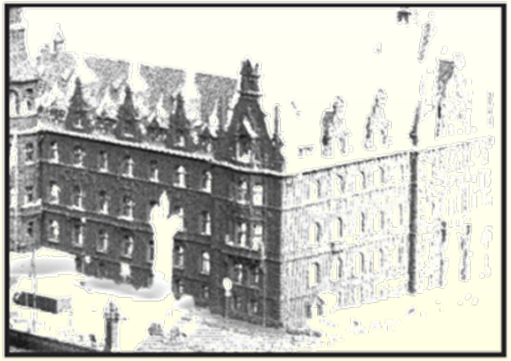
A. Repeated Pattern Detection

[M. Pauly, N. Mitra, J. Wallner. L. G., H. Pottmann]

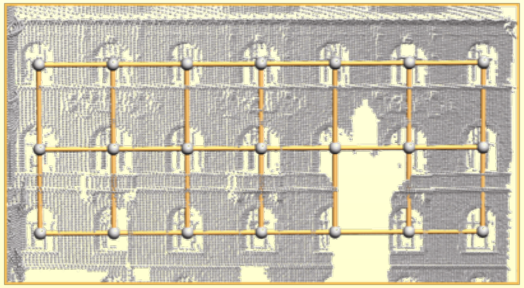


Structure Discovery

- Discover regular structures in 3D data, without prior knowledge of either the pattern involved, or the repeating element
- Algorithm has three stages:
  - Transformation analysis
  - Model estimation
  - Aggregation



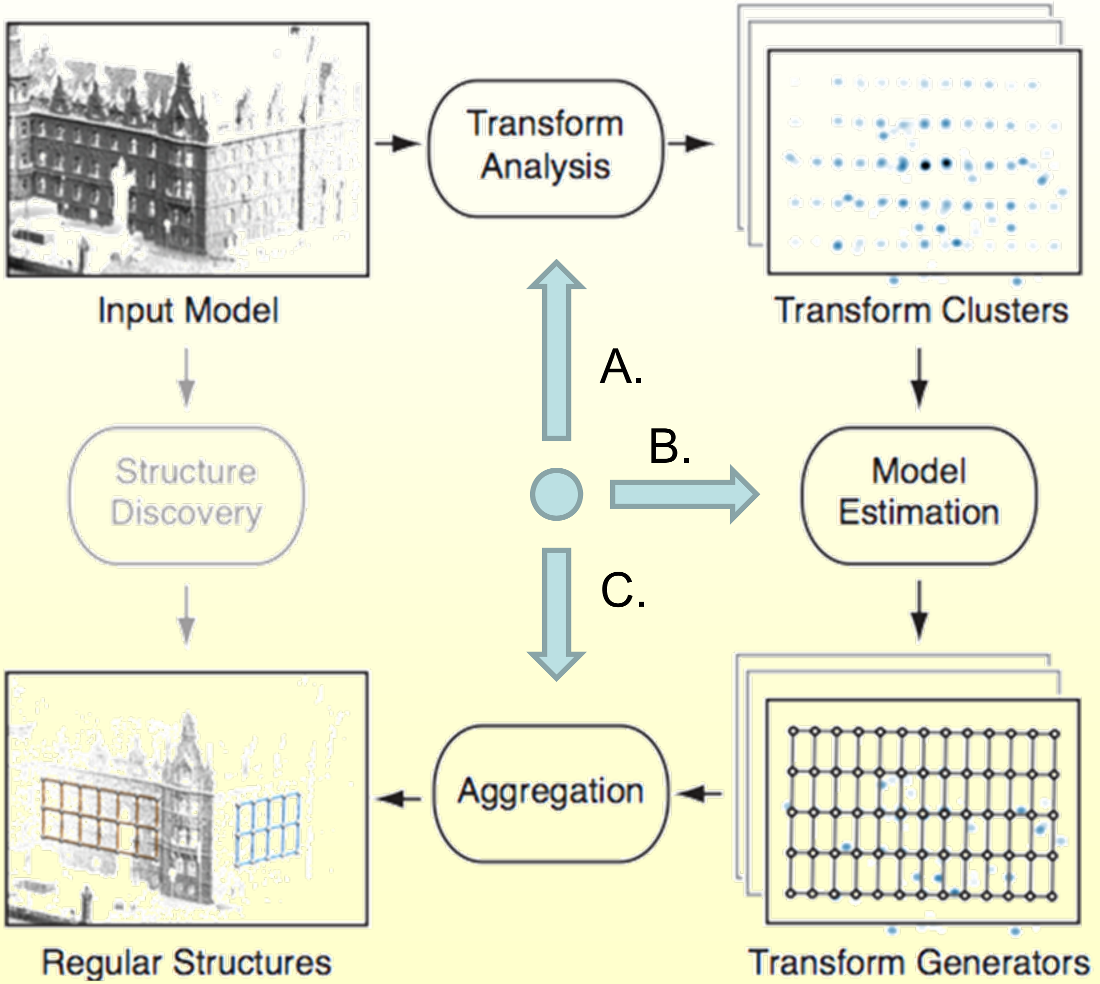
Input Model



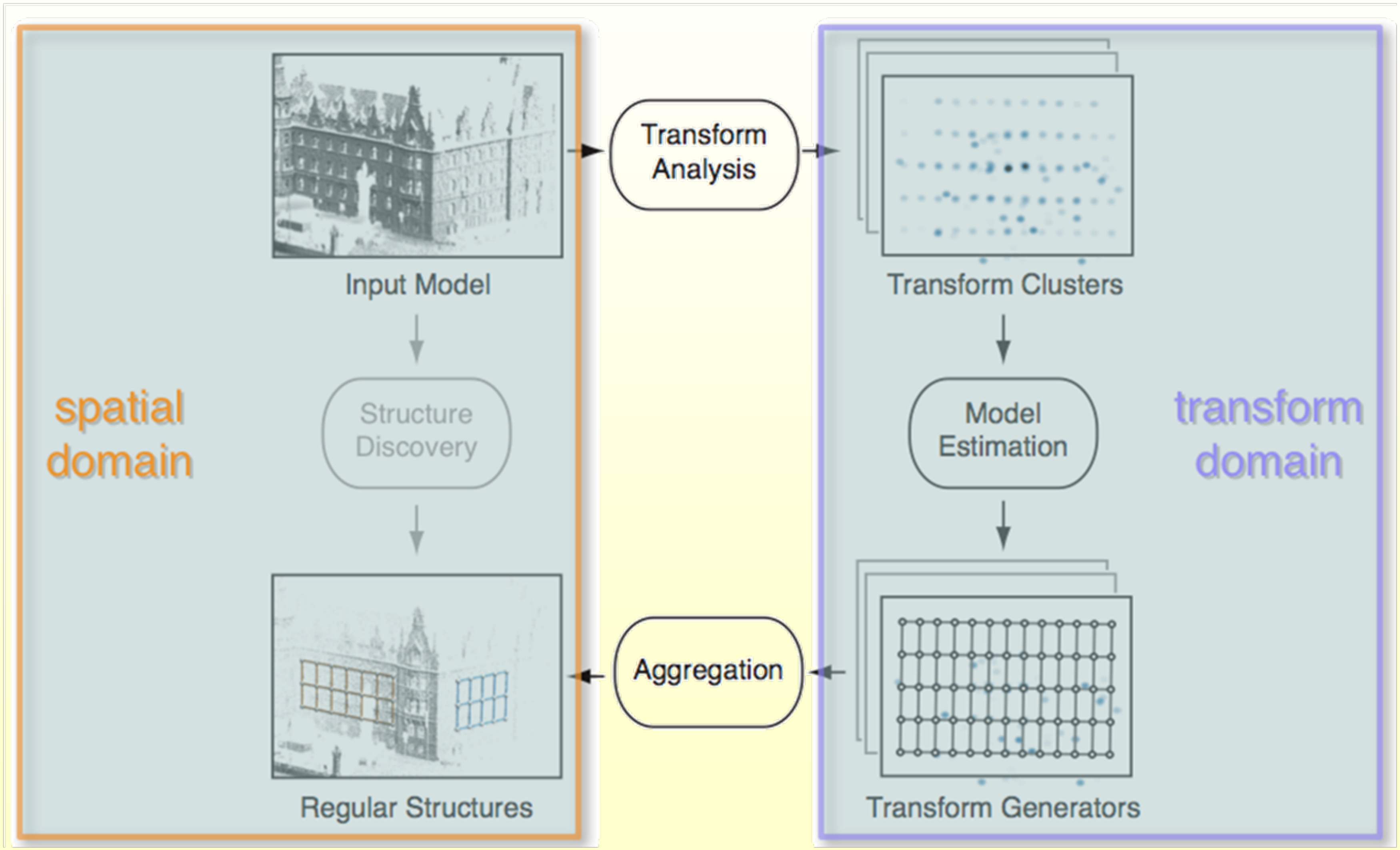
Regular structure

Challenges: joint discrete and continuous optimization, presence of clutter and outliers

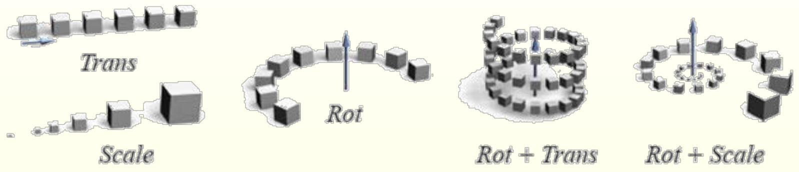
Algorithm Overview



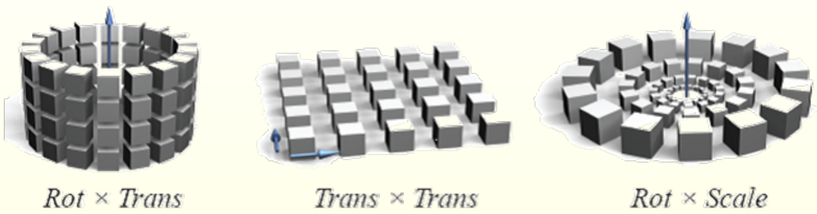
Algorithm Overview



Repetitive Structures



1D structures

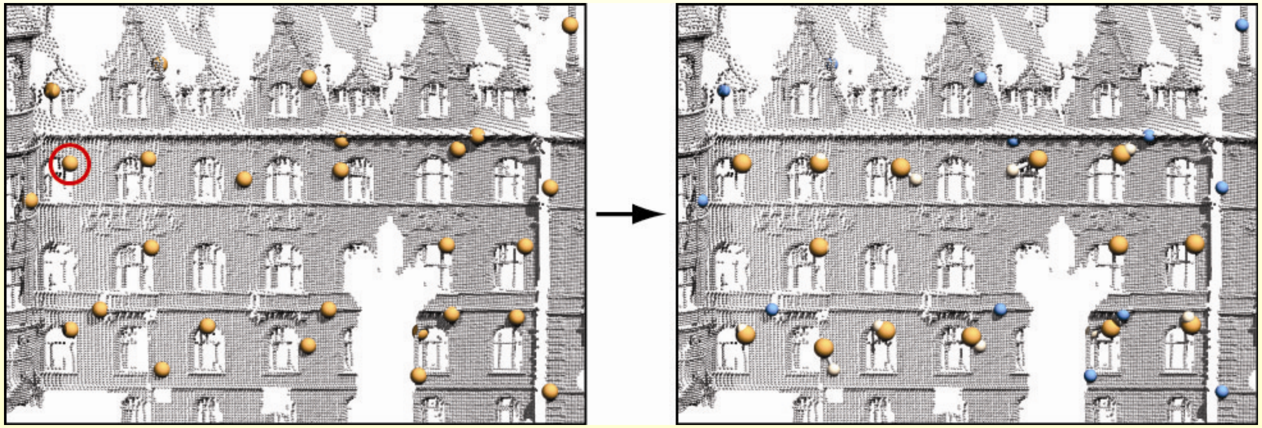


2D structures

Regular structures:
rotation + translation + scaling $\rightarrow$ any commutative combinations in the form of 1D, 2D grid structures

Similarity Sets

Compare all pairs of small patches, using local shape descriptors

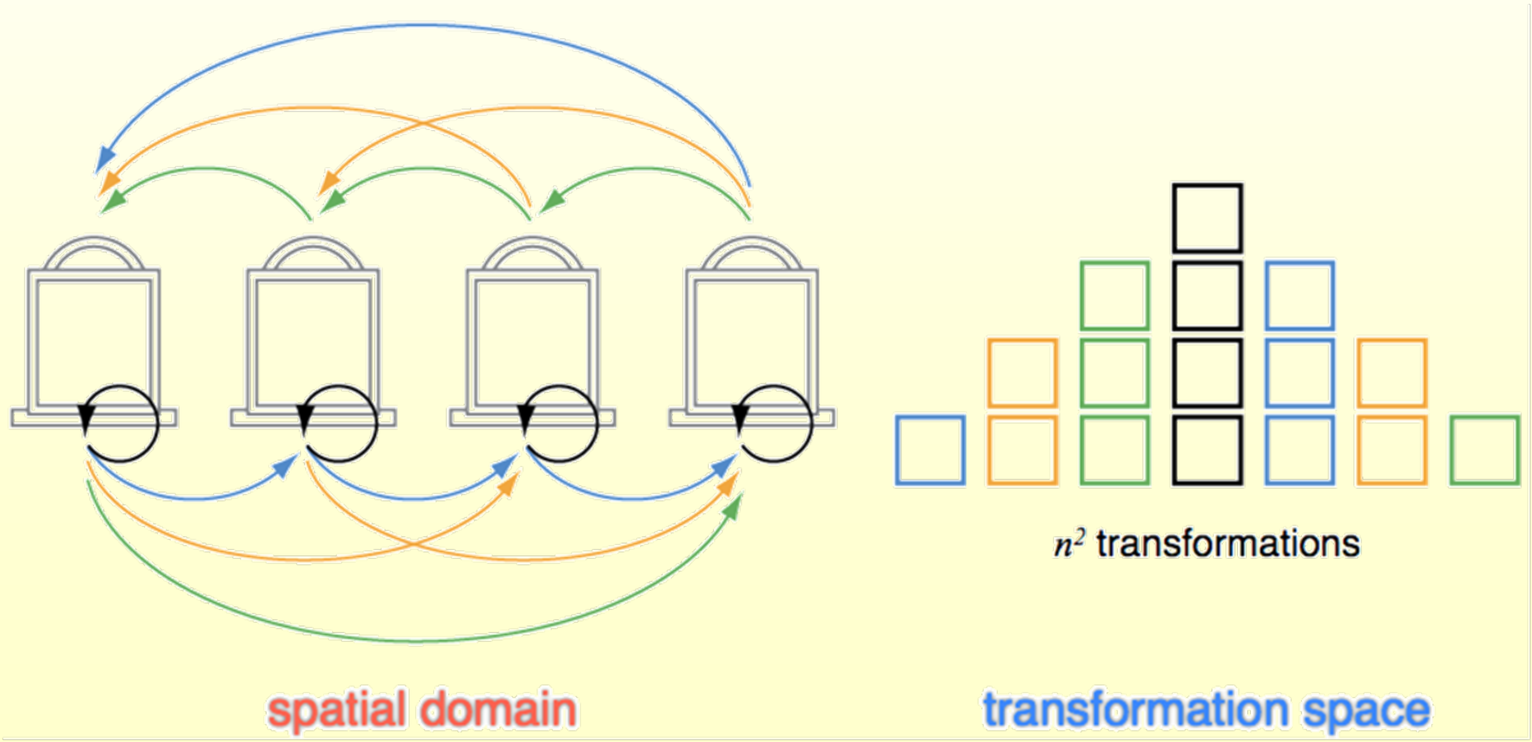


Based on shape descriptors alone

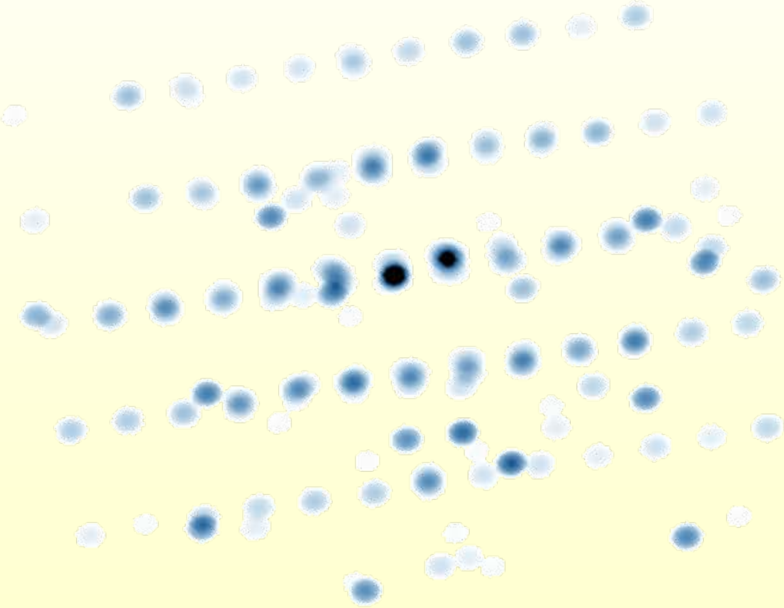
Pruned, after validation w. geometric alignment

Transform Analysis

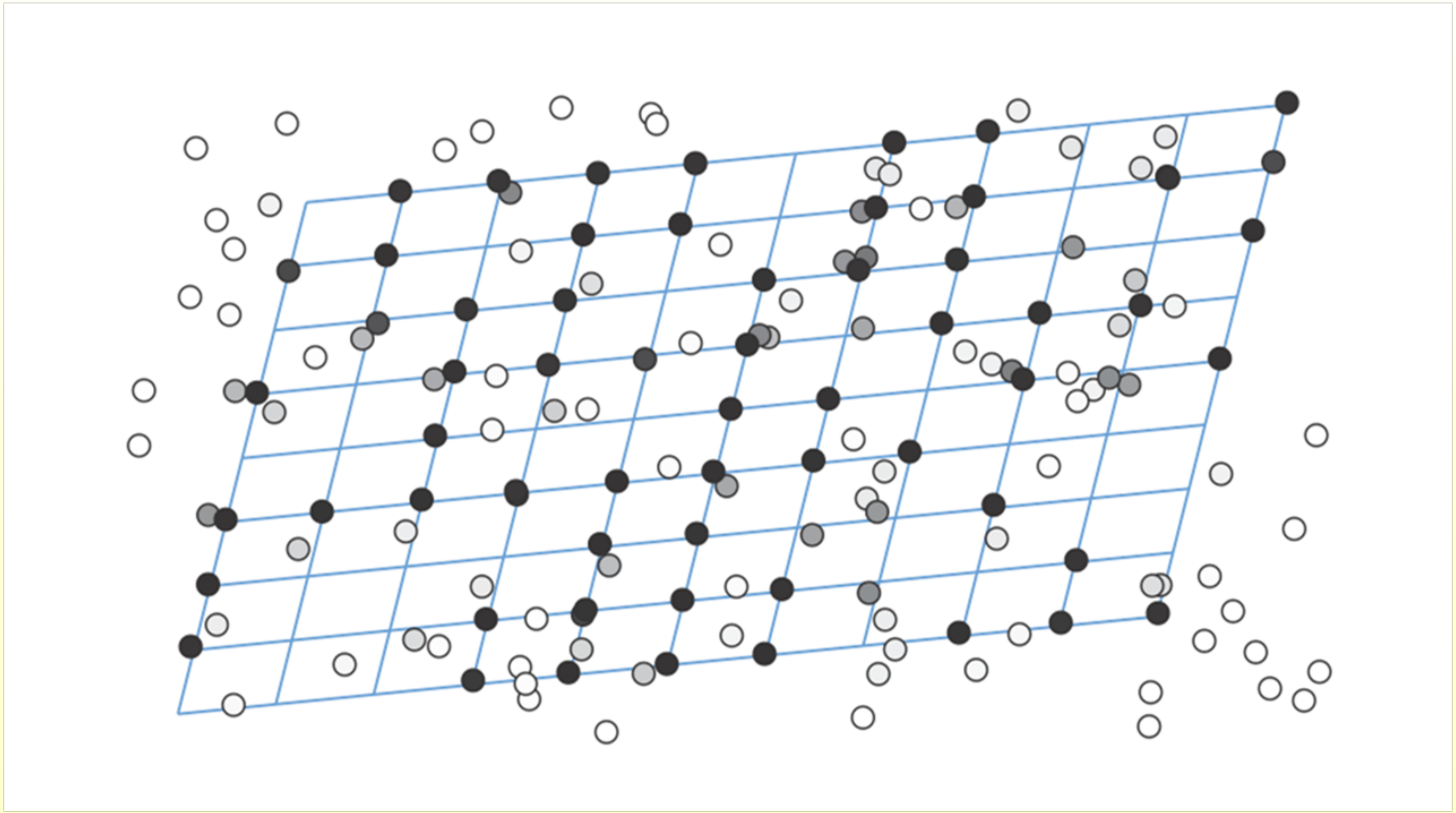
- Regularity in the spatial domain is enhanced in the transform domain



Density Plots in Transform Space



Model Estimation: Where is the Grid?



Grid Fitting with Clutter and Outliers

Clusters in
transform space

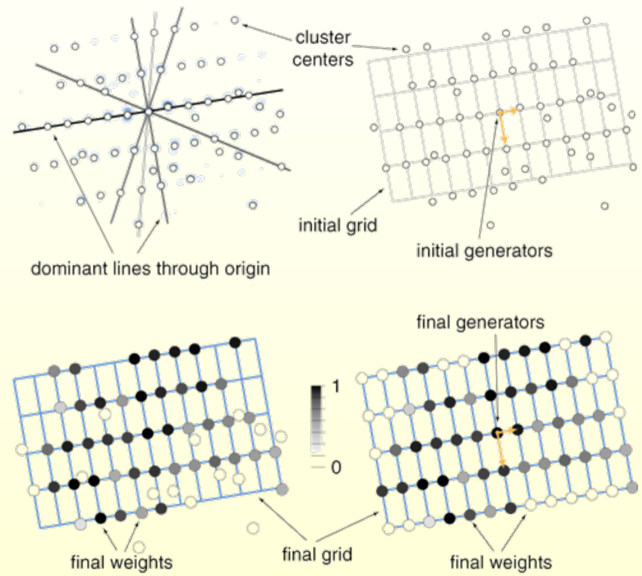
$$\vec{g}_1, \vec{g}_2, \{\alpha_{ij}\}, \{\beta_k\} = \underset{\vec{g}_1, \vec{g}_2, \{\alpha_{ij}\}, \{\beta_k\}}{\operatorname{argmin}} E$$

$$E = \gamma(E_{X \to C} + E_{C \to X}) + (1 - \gamma)(E_\alpha + E_\beta)$$

$$E_{X \to C} = \sum_i \sum_j \alpha_{ij}^2 \|\vec{x}_{ij} - \vec{c}(i,j)\|^2$$

$$E_{C \to X} = \sum_{k=1}^{|C|} \beta_k^2 \|\vec{c}_k - \vec{x}(k)\|^2$$

$$E_\alpha = \sum_i \sum_j (1 - \alpha_{ij}^2)^2 \qquad E_\beta = \sum_k (1 - \beta_k^2)^2$$



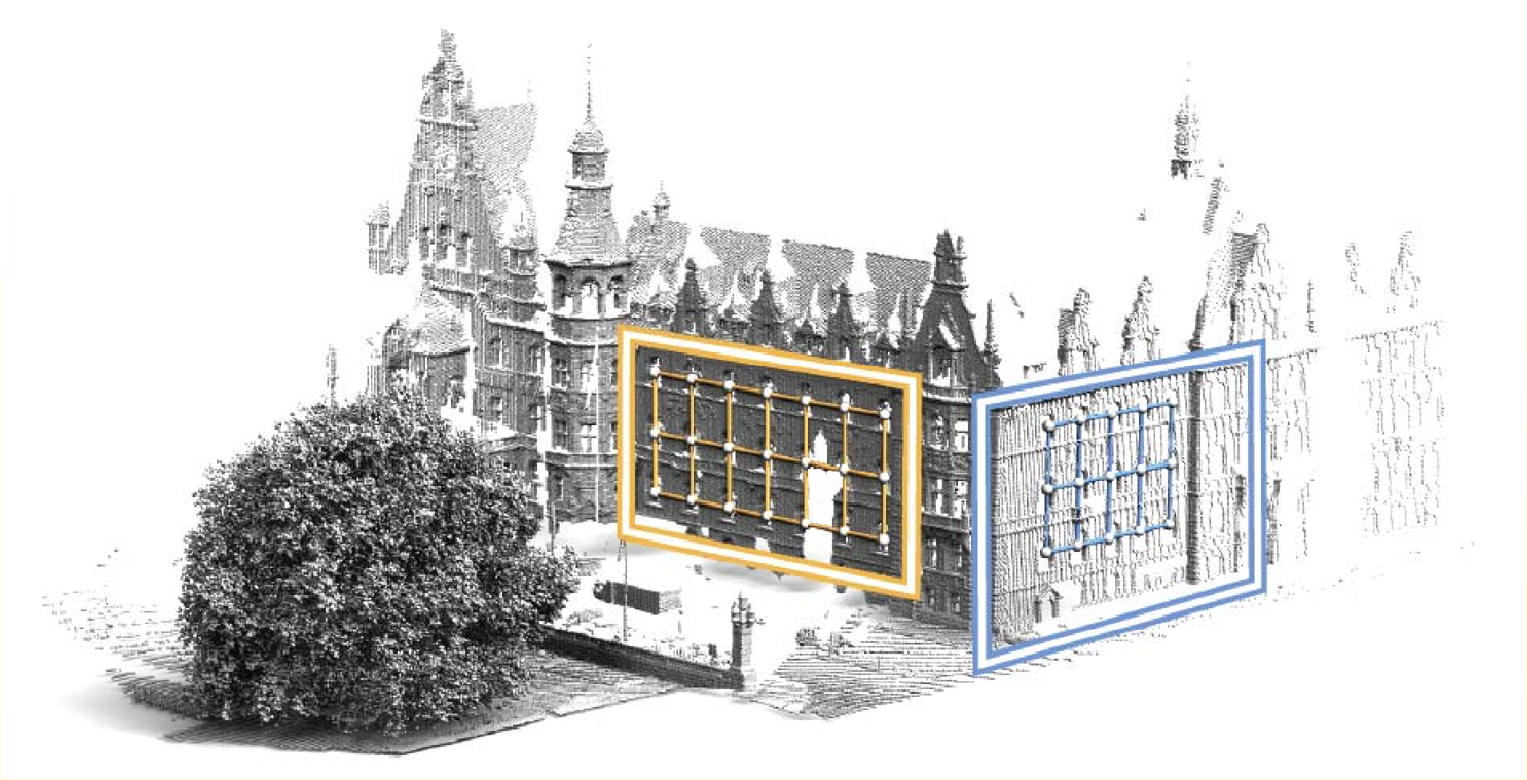
α β

X = grid
C = transform cluster

Aggregation

- Once the basic repeated pattern is determined, we simultaneously (re-)optimize the pattern generators and the repeating geometric element it represents, going back to the original 3D data
- We inteleave
  - region growing
  - re-optimization of the generating transforms of the pattern by performing simultaneous registrations on the original geometry

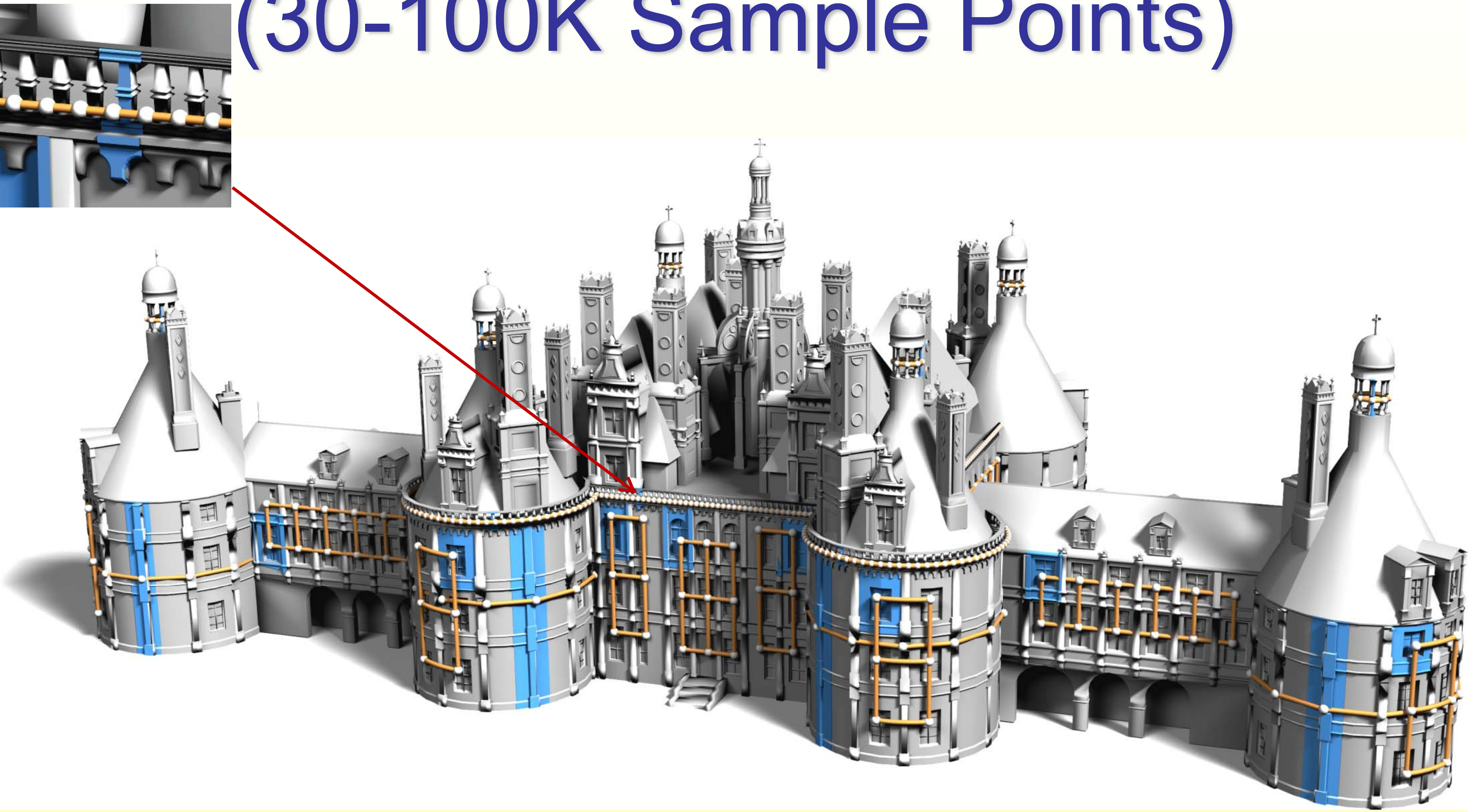
Scanned Building Facade



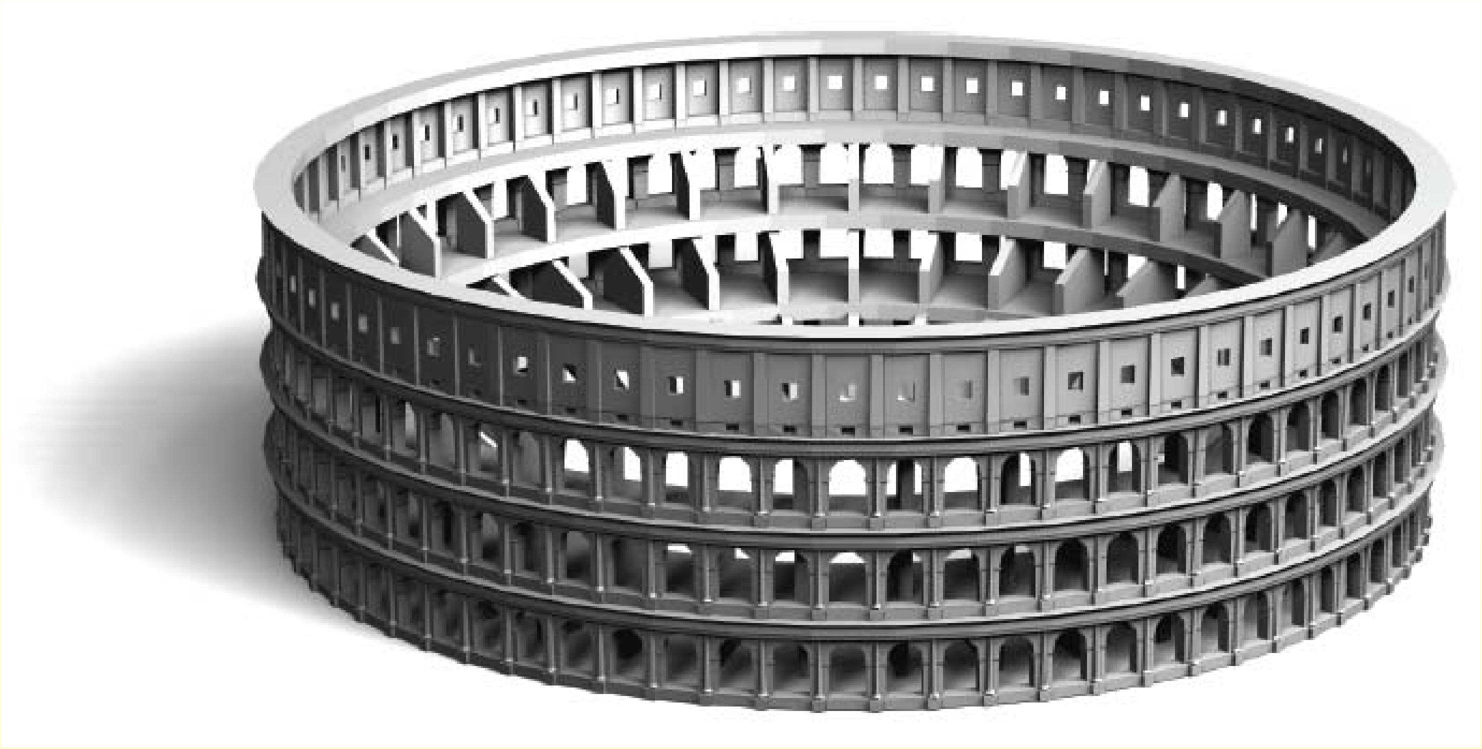
Output:

- Golden: 7x3 2D grid
- Blue: 5x3 2D grid

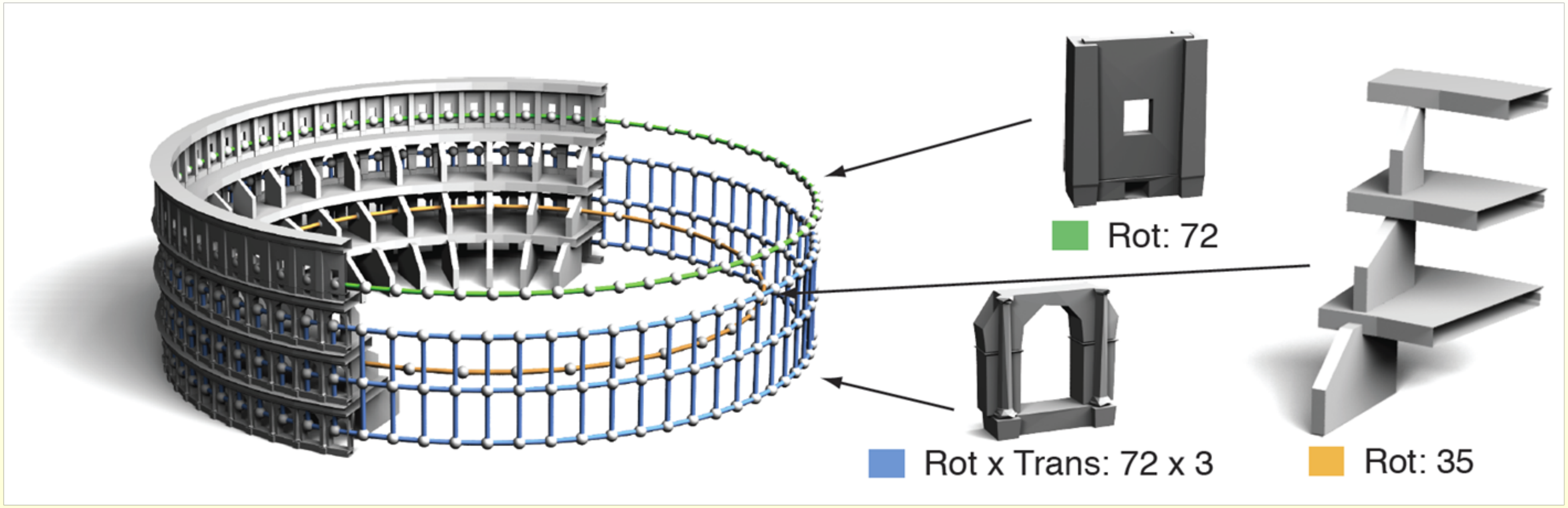
Back to Chambord (30-100K Sample Points)



Amphitheater

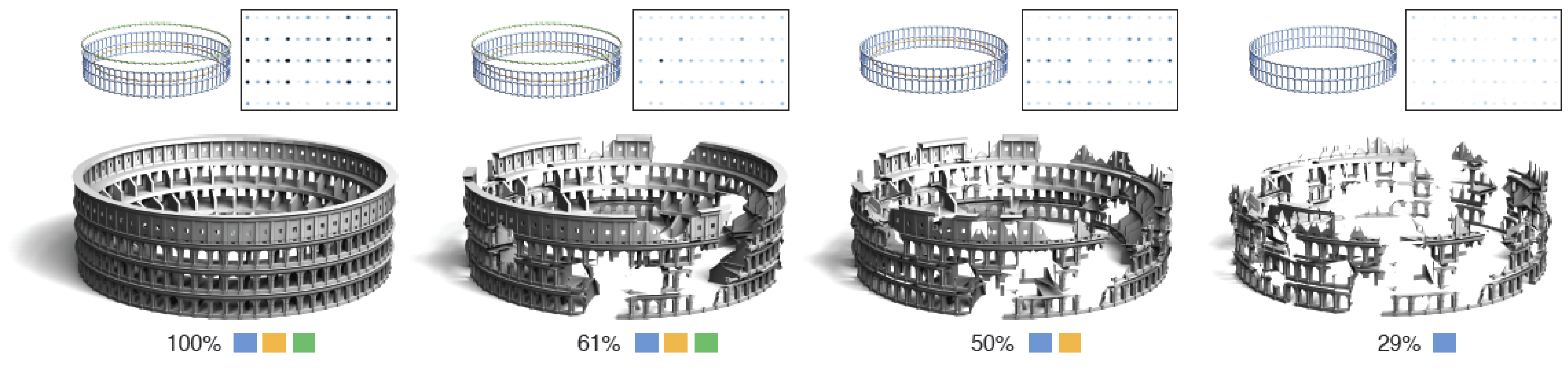


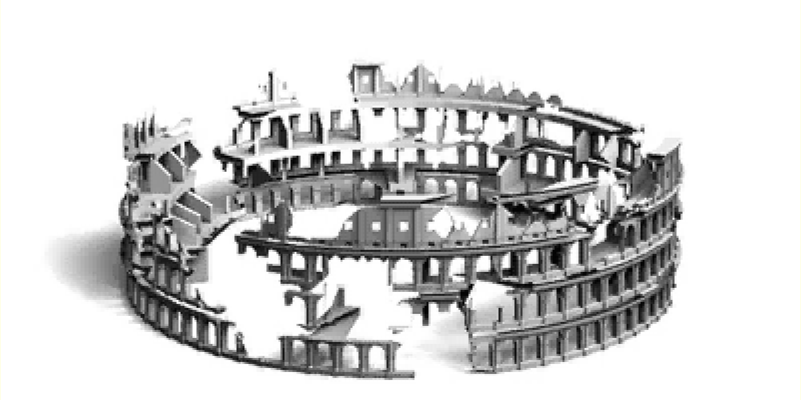
Amphitheater



Output: 3 grids + associated patches

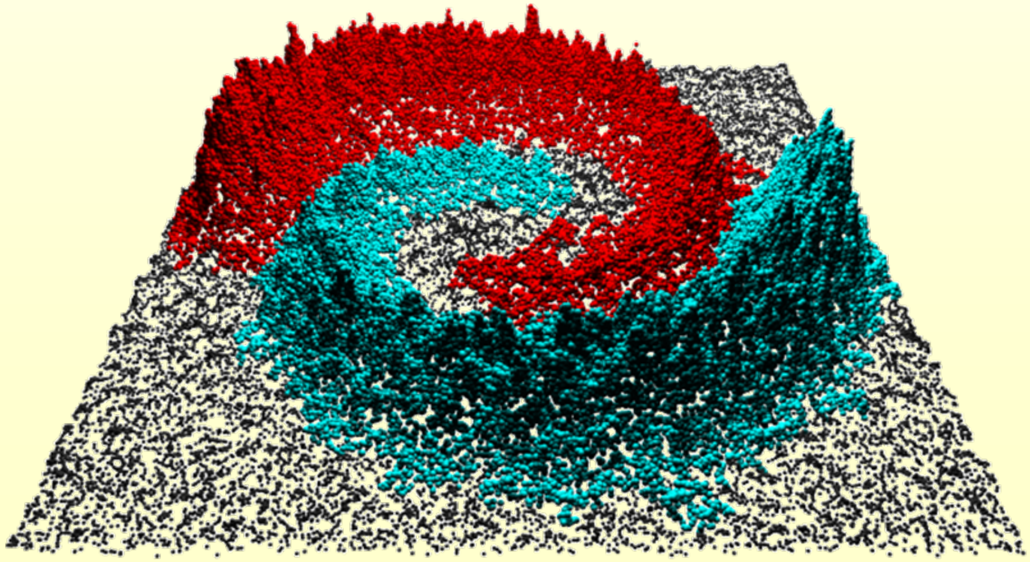
Robustness to Missing Data





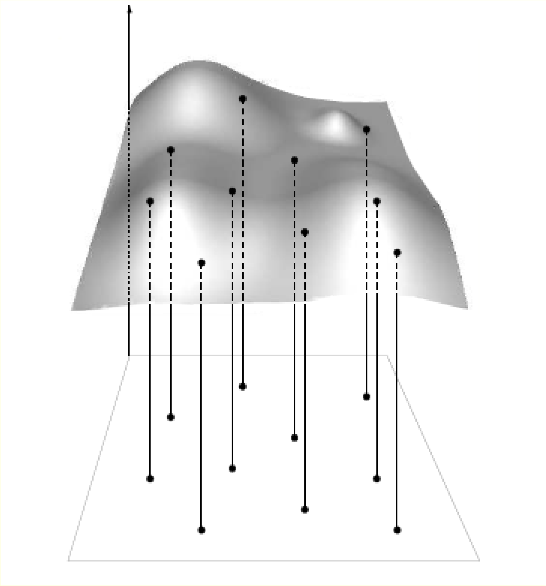
III. Scalar Field Analysis over Riemannian Spaces

[F. Chazal, L. G., S. Oudot, P. Skraba]



Scalar Field Analysis

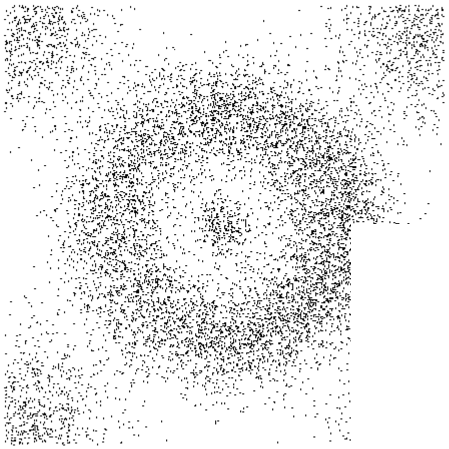
- We are given a Riemannian space *X* and a Lipschitz function *f* over *X*. We know *X*, *f* only through samples. We can access
  - the distances between the samples
  - the values of *f* at the samples



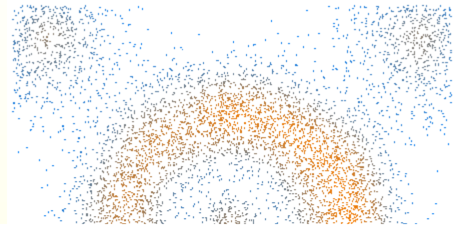
- We want to analyze the shape of *f:*
  - Detect significant peaks/valleys
  - Detect changes in the sublevel sets of *f*
- We approach the problem through *persistent homology*

Clustering Density Functions

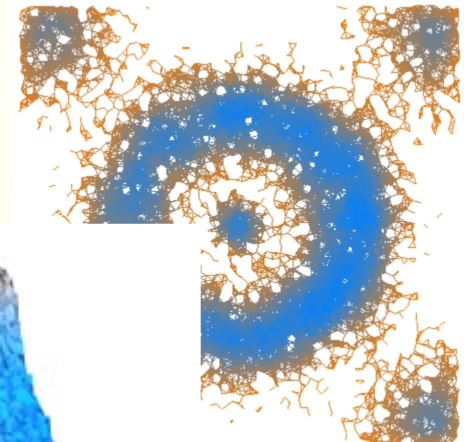
Point cloud

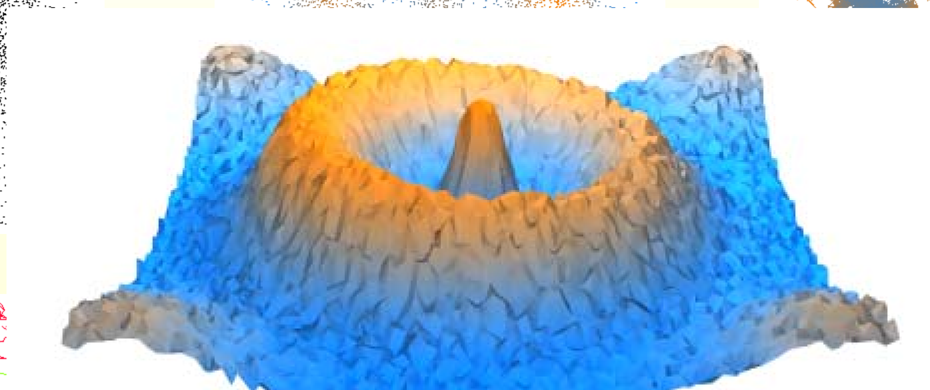


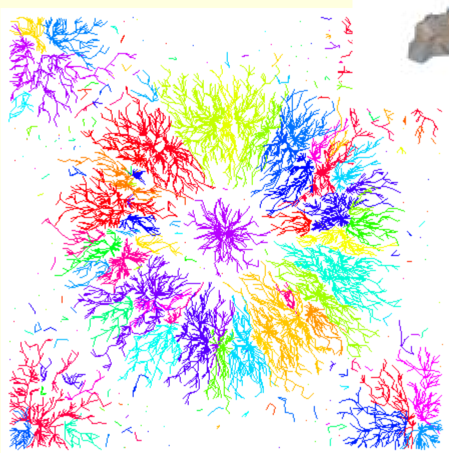
Density estimation



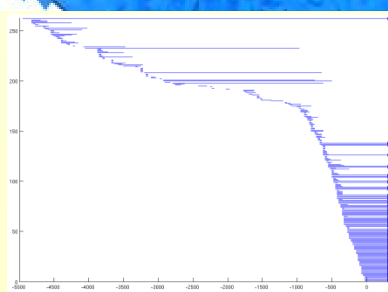
Rips filtration



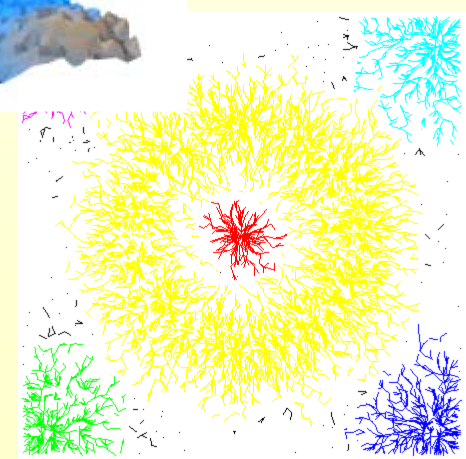




Initial basins/clusters



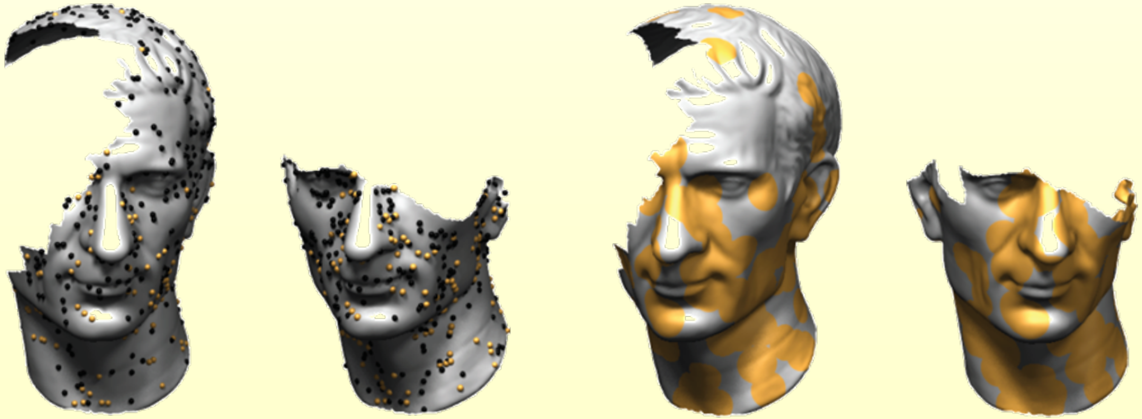
Persistence barcode



Final basins/clusters

IV. Fingerprints for Distributed Data Analysis

[M. Pauly, J. Giesen, N. Mitra, L. G.]



Probabilistic Fingerprints

partial
similarity

probabilistic
fingerprint

Shapes are never
directly compared

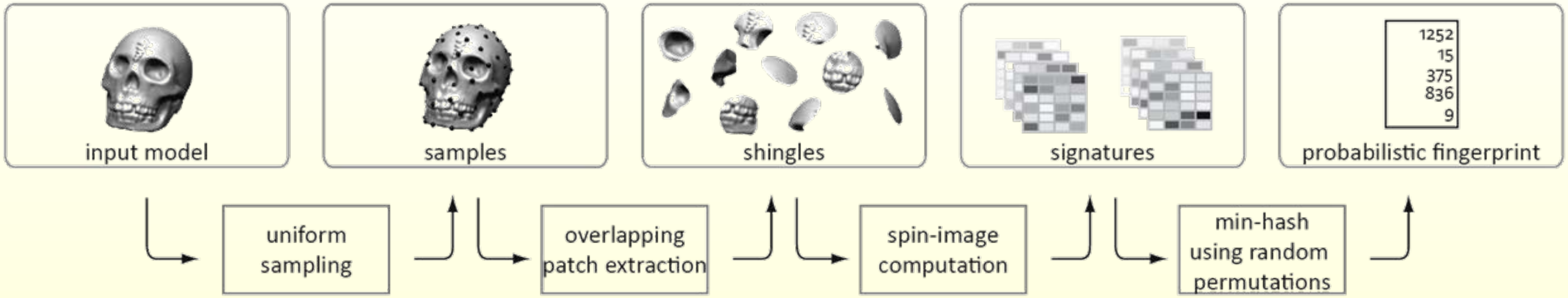
probabilistic
fingerprint

compact

compare

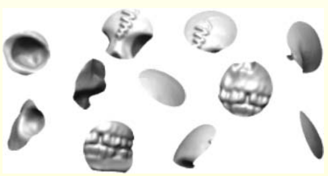
independent

Fingerprint Pipeline



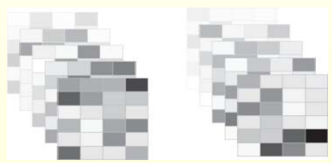
Data Reduction

quantization | min hashing



Shingles

→



Signatures

→

9501230
8913762
6154738
7919176
9218405

8936813
5790099
3529138
1987604
2722746

Descriptors

→

1252
15
375
836
9

Fingerprint

100k | 100k | 100k | 1k

set size remains constant | set reduction

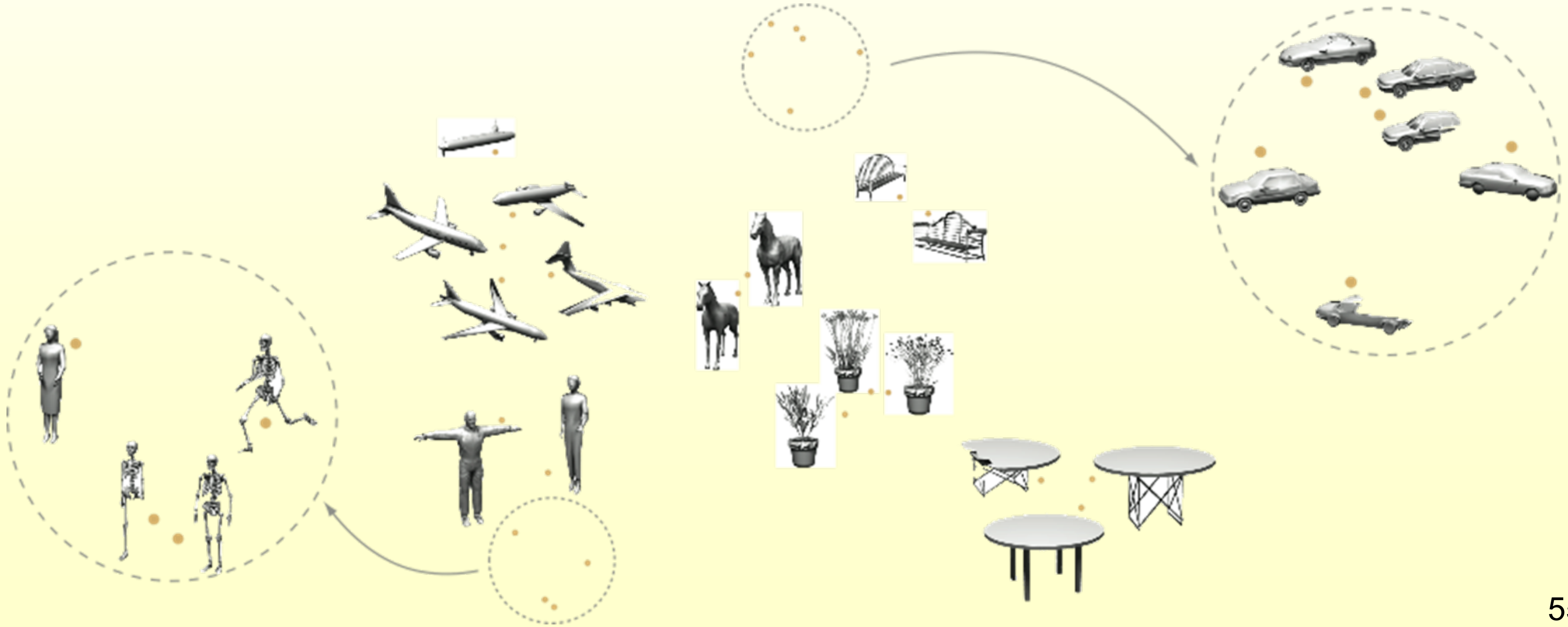
Applications

- Resemblance between partial scans

	53.9	59.8	35.1
55.2		21.5	24.3
63.1	17.9		30.9
39.5	19.3	35.5	

Applications

- Shape distributions



Challenge: From 3-D to Any-D

- Can these techniques be applied to higher-dimensional settings (low-d data sets in high-d ambient space)?
  I. How do we estimate good local descriptors for high-dimensional data?
  II. What if the data is sparse?
  III. Are there “structure-preserving” low-d projections and embeddings?
- Can we handle dynamic data changes, streaming data sets, etc?

Challenge: Exploiting Structure for Interaction

- Structure → User
  - We can extract interesting parts of the data, or relationships between parts, or regular patterns present in the data
  - But how can one display effectively discovered structure in higher dimensions?
- User → Structure
  - How should the user be able to influence the structure discovery process?
  - How can the user
    - seek additional data to confirm structure?
    - manipulate data to enhance structure?

FODAVA Contribution

- If we succeed, we will have a set tools for data analysis that
  - have a rigorous mathematical foundation in geometry and topology
  - efficiently discover intrinsic structures in data
  - can deal in a lightweight fashion with large scale, distributed data sets
  - integrate well with techniques for visualization and interactive exploration
  - can be of interest to other communities within computer science and applied mathematics