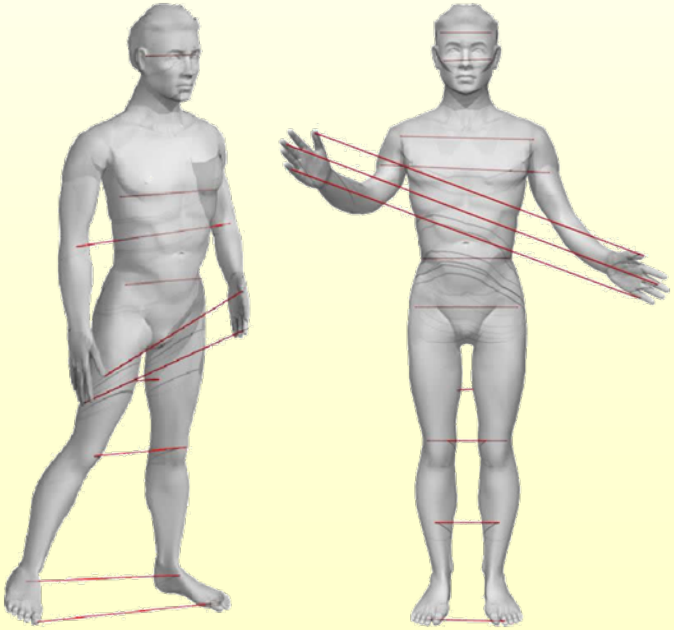
Detection of Symmetries and Repeated Patterns in 3D Point Cloud Data

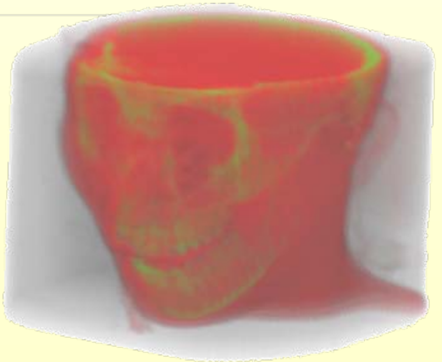
Leonidas J. Guibas
Computer Science
Stanford University

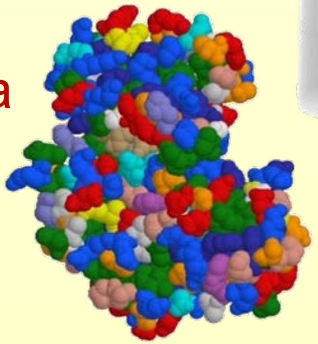


MMDS 2008

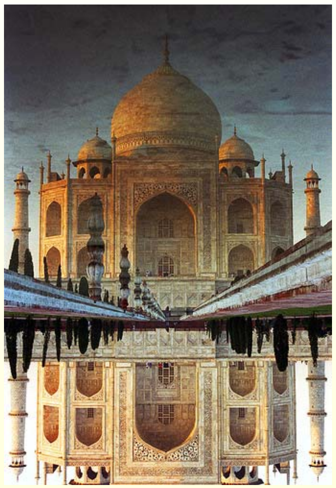
3D Digital Shape Modeling

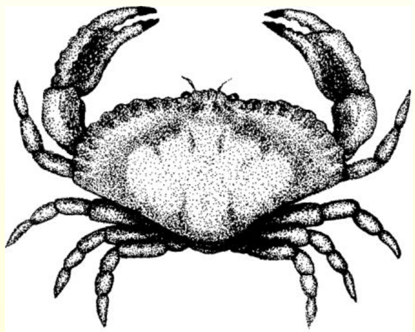
- More and more shapes around us are being digitized
  - shapes of manufactured objects (CAD models)
  - 3-D scanning for acquired geometry
  - shapes of organs in our bodies
  - shapes of molecules (proteins)
- We need tools for analyzing and processing digital geometry
  - images, audio → video → geometry data
- With many acquisition technologies, the initial data is a point cloud (PCD)
- We want to develop techniques for extracting structural regularities in such data

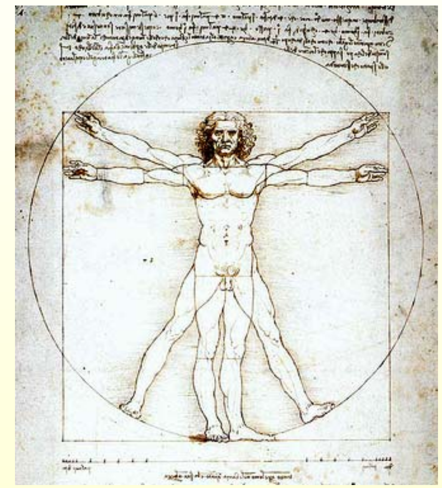




Symmetries and Regular Patterns In Natural and Man-Made Objects





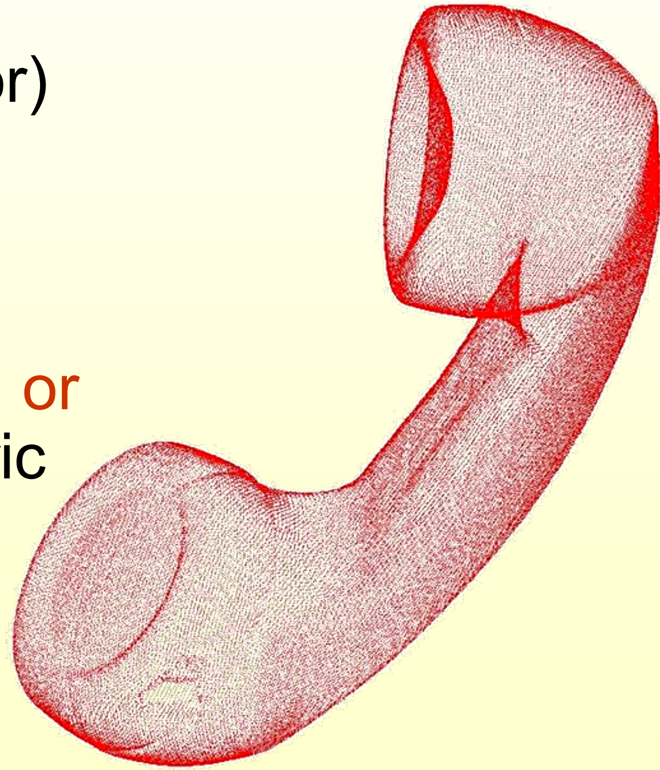


"Symmetry is a complexity-reducing concept [...]; seek it everywhere.
Alan J. Perlis

Point Cloud Data (PCD) Pose Particular Challenges

- PCD = “point cloud data”
  - unorganized collection of points sampled from the surface (or interior) of an object, with noise added
  - typical output of a 3-D scanning process
- no connectivity information or manifold or mesh structure ⇨ hard to use geometric methods directly
- no regular sampling

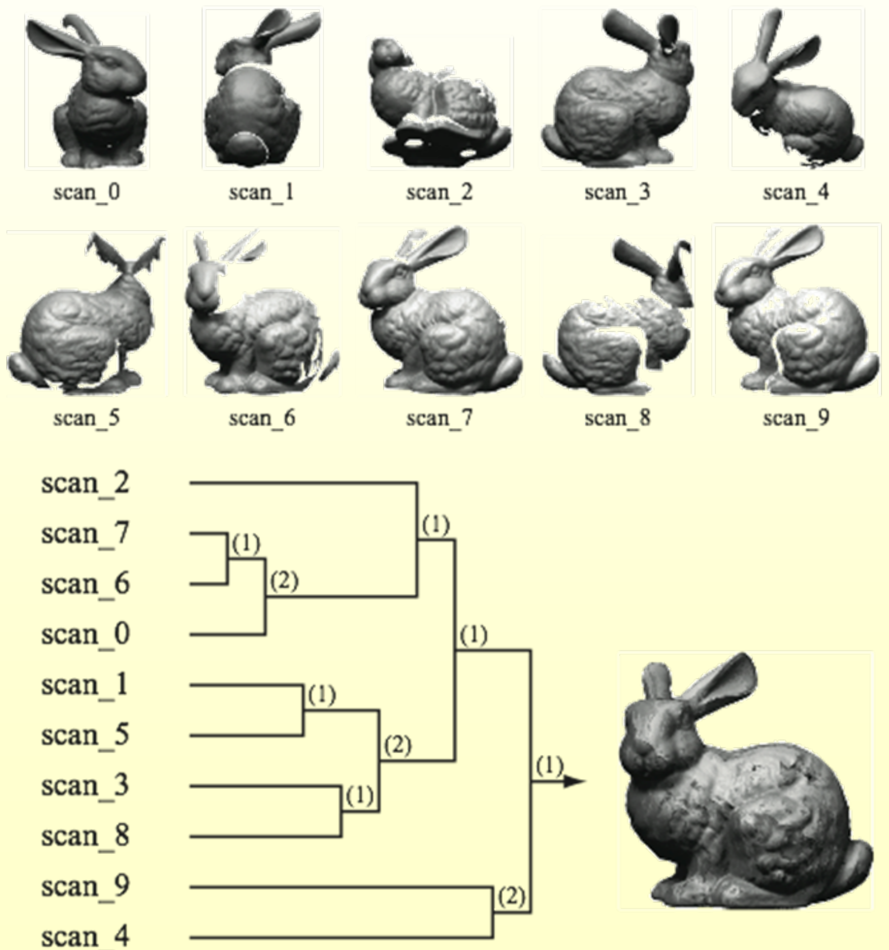
⇨ hard to use signal processing tools



Distributed Data Sets

- Data sets of interest may be distributed over a network
- May be massive
- May have different owners

- How to decide when data sets should be, or can be, fused, compared, etc?



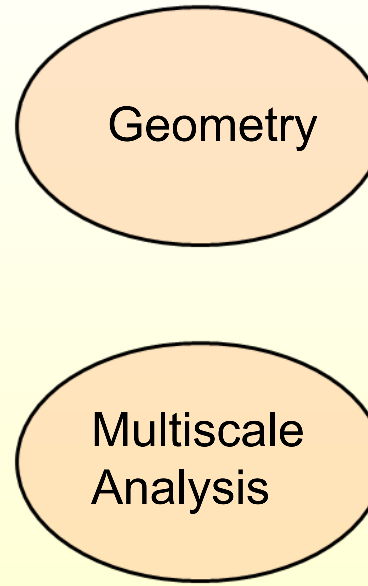
Geometric Structure Extraction as a Paradigm for Data Analysis

- All of science and engineering is becoming data rich
  - massive data coming from sensors
  - massive data coming from simulations
- Such data from physical processes is often in the form of unorganized point clouds
- Machine learning is fundamentally based on fitting functions to data (regression, classification)
- An alternative approach can be comparing data to itself, or to other data of the same type

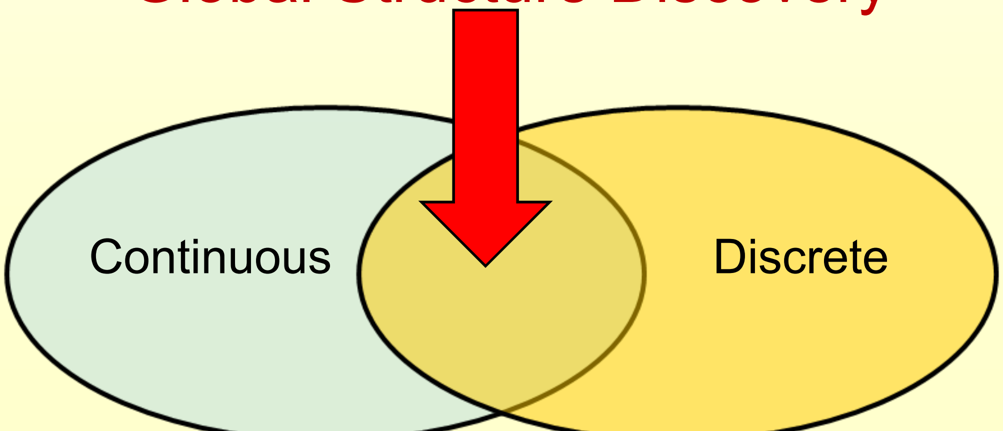
$$\text{Physical Laws} \overset{?}{=} \text{Symmetries}$$

Computational Symmetry

I. Symmetry Extraction and Symmetrization
II. Distributed Congruence Discovery
III. Repeated Pattern Detection



Global Structure Discovery



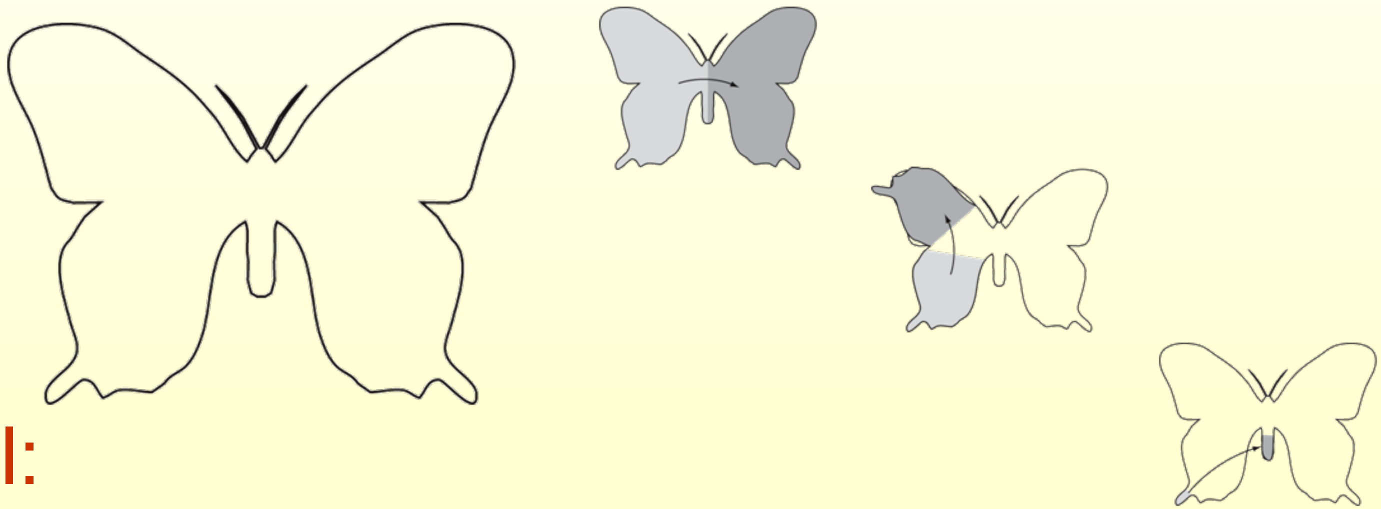
I. Symmetry Extraction and Symmetrization

[Mitra, G., Pauly, Siggraph ’06, Mitra , G., Pauly, Siggraph ’07]

Partial/Approximate Symmetry Detection

Given:

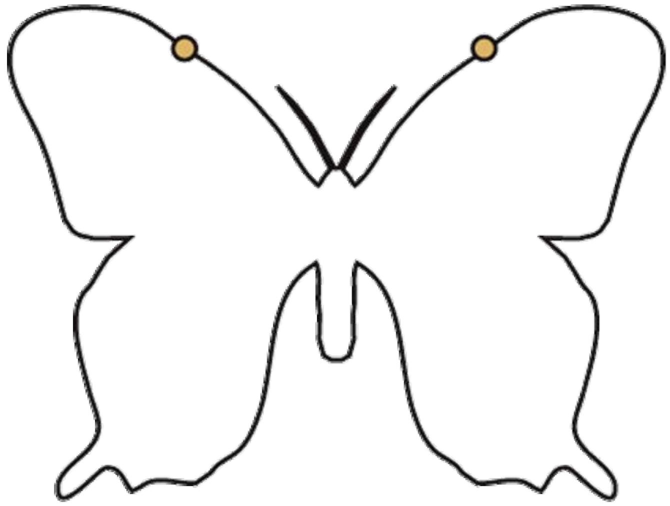
Object/shape (represented as point cloud, mesh, ...)



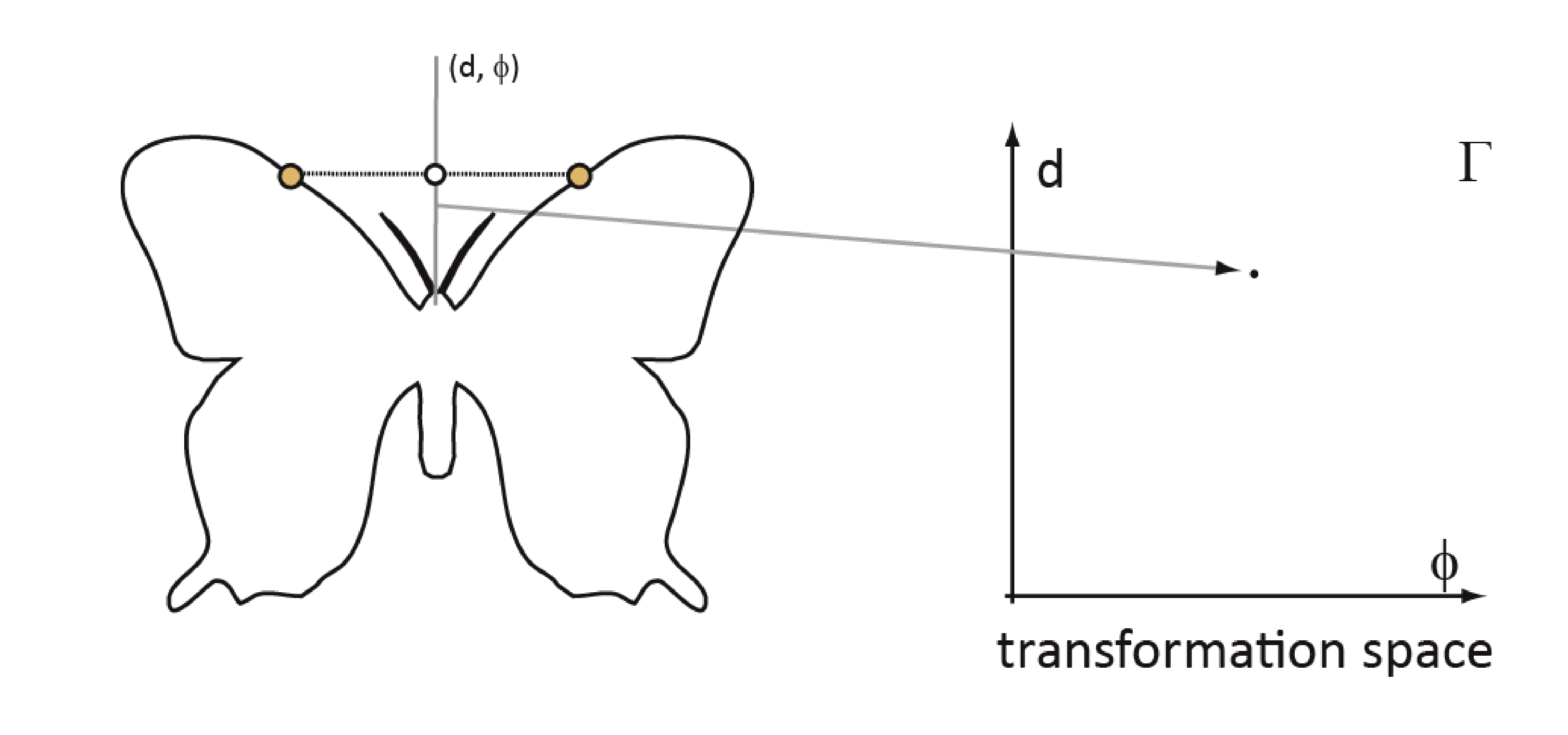
Goal:

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

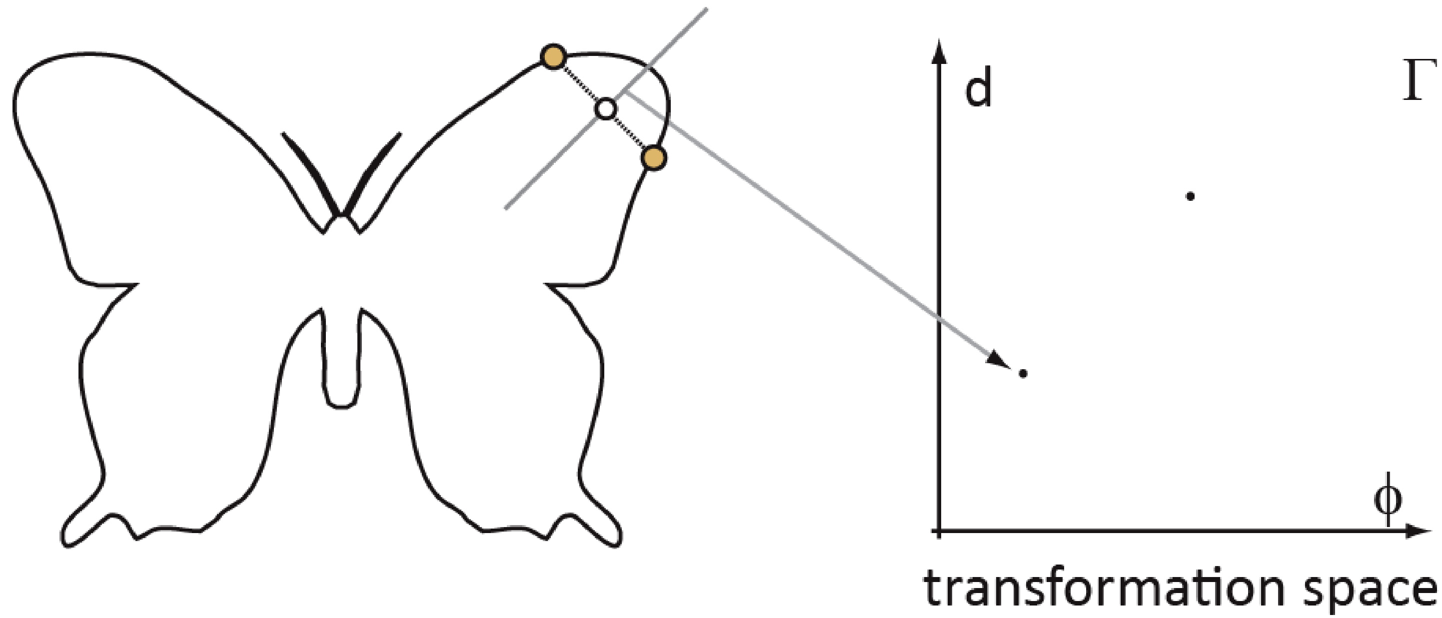
An Example: Reflective Symmetry



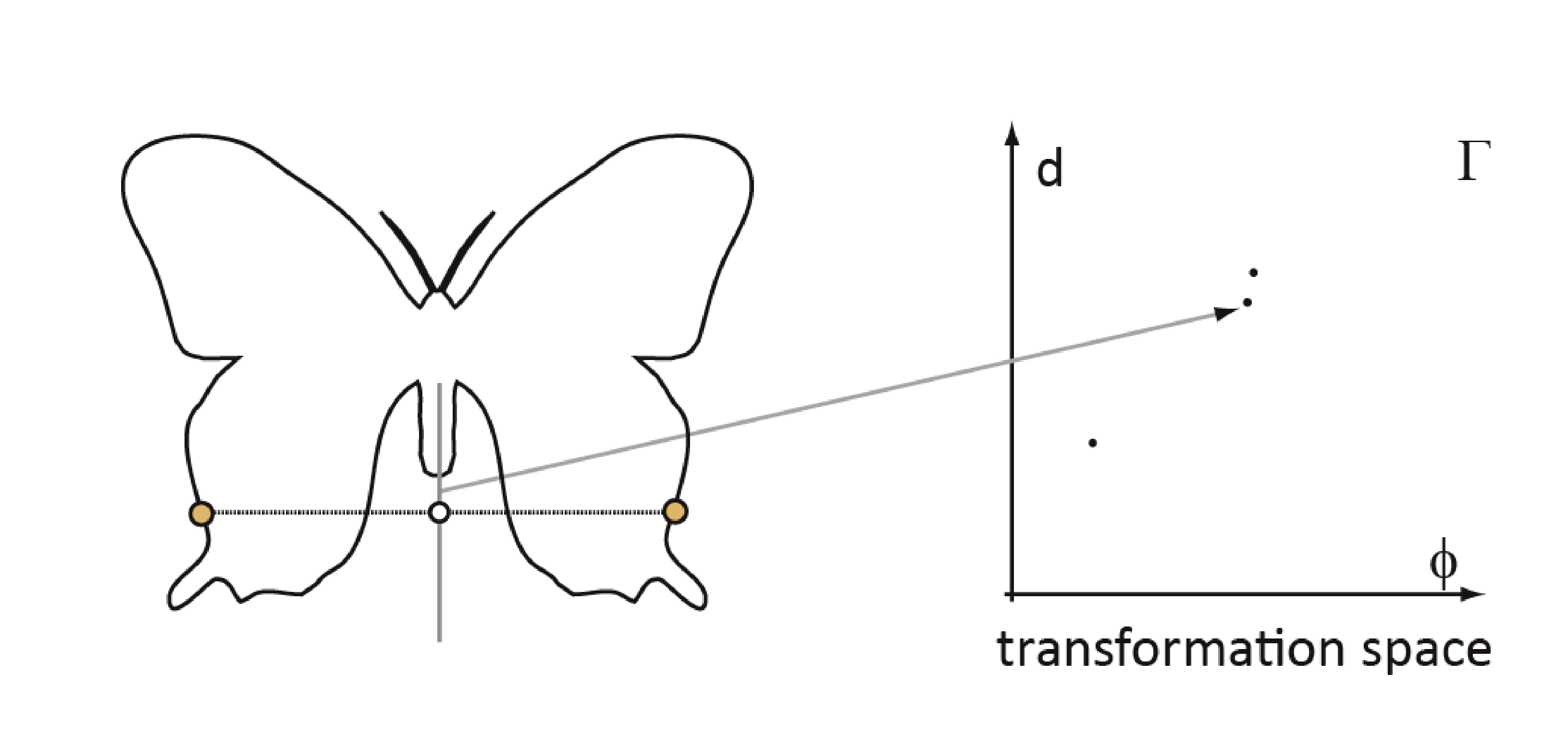
Reflective Symmetry : A Pair Votes



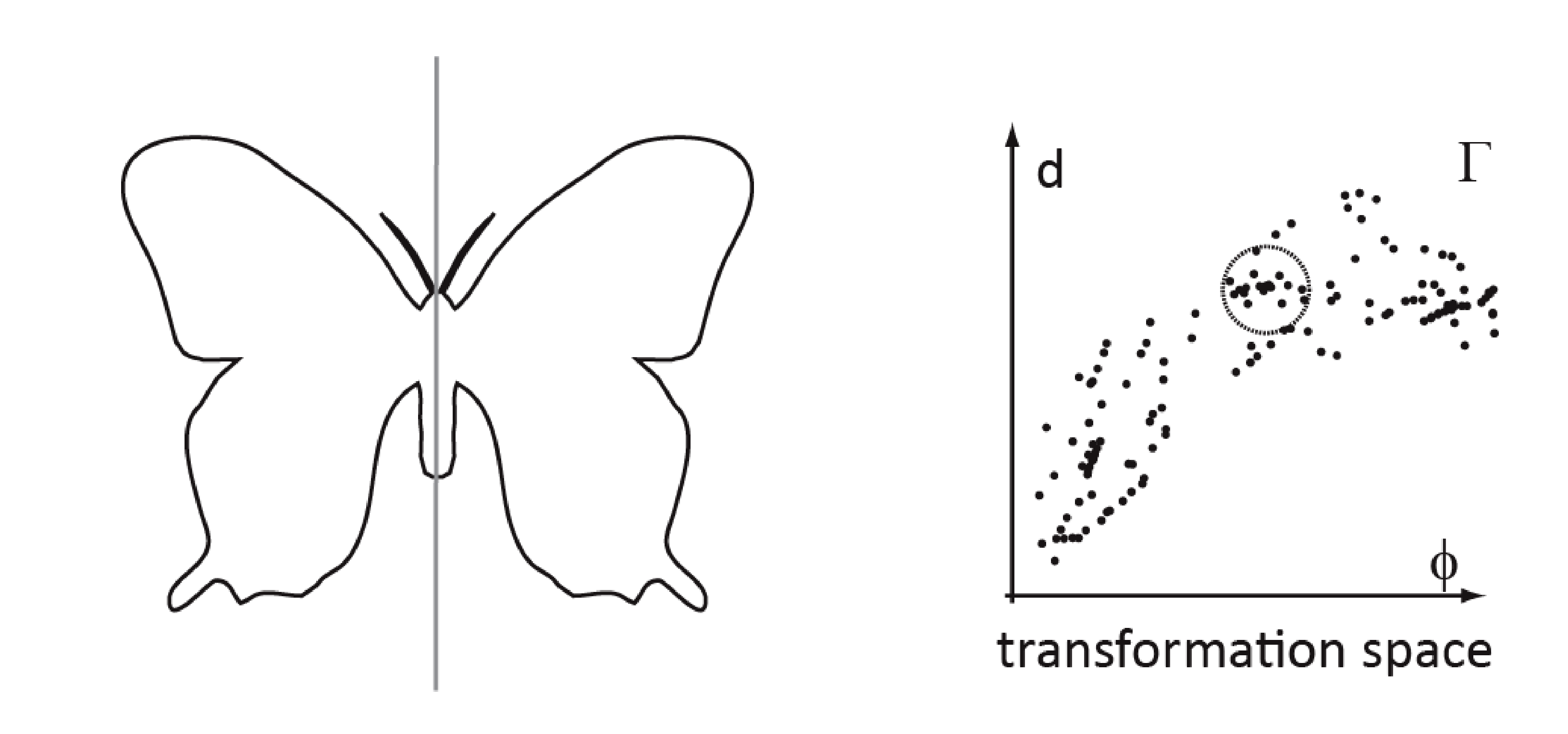
Reflective Symmetry : Voting Continues



Reflective Symmetry : Voting Continues

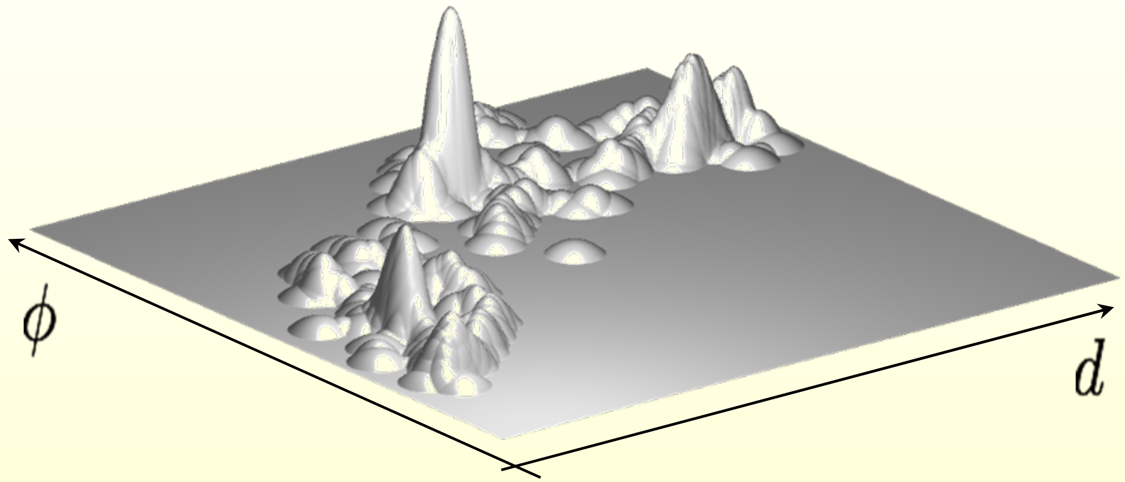


Reflective Symmetry : Largest Cluster



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

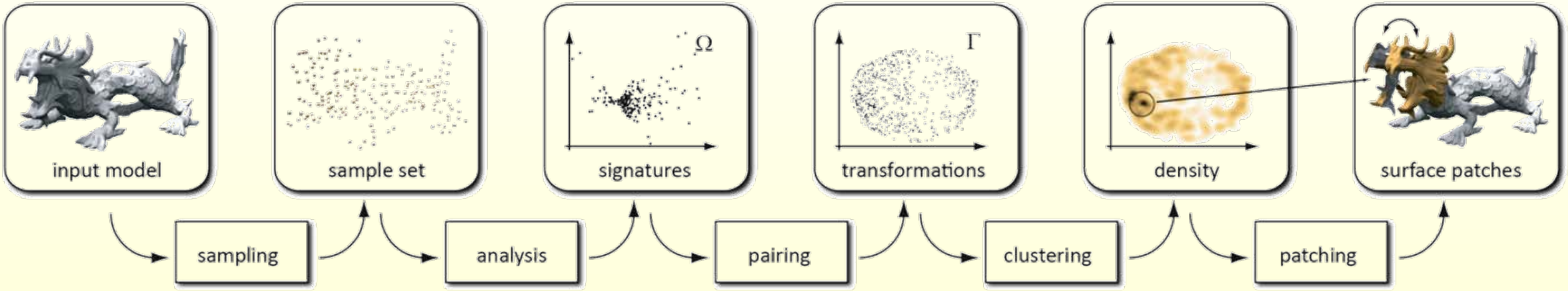
A Typical Density Plot



height of cluster $\rightarrow$ *extent* of approximate symmetry

spread of cluster $\rightarrow$ *deviation* from exact symmetry

Pipeline



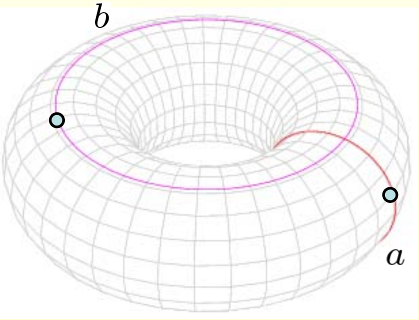
input model
sample set
signatures
Ω
transformations
Γ
density
surface patches
sampling
analysis
pairing
clustering
patching

Pruning: Local Signatures

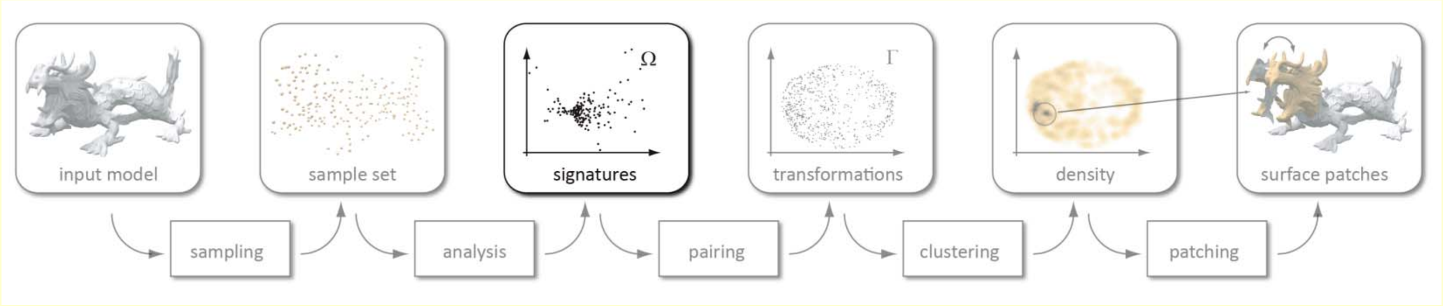
- Local signature → invariant under transforms
- Signatures disagree → points don't correspond

Use (κ_1, κ_2) for curvature based pruning

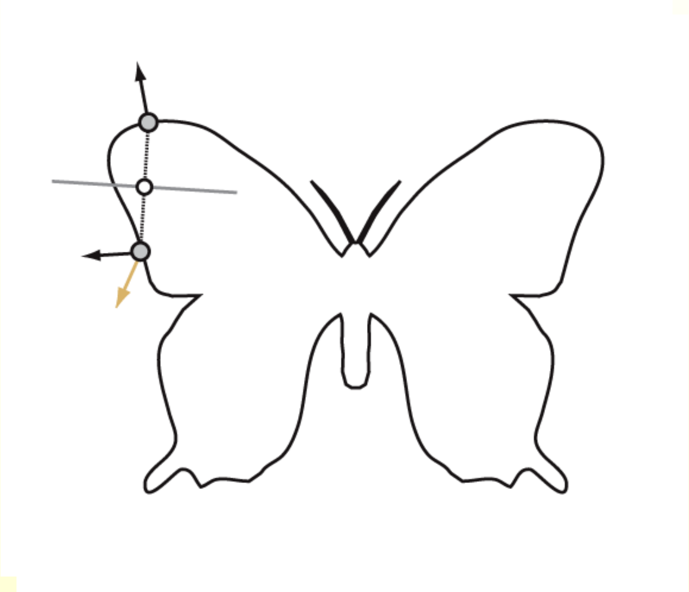
$(1/(a+b), 1/a)$

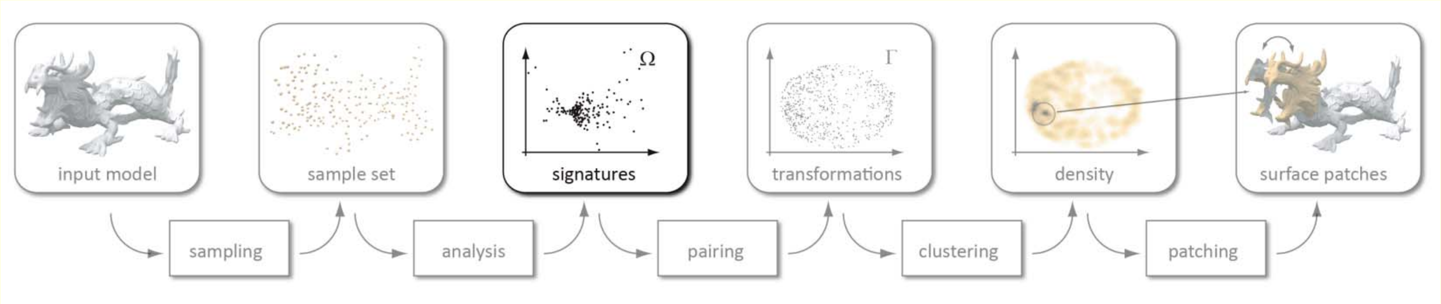


$(0, 1/a)$

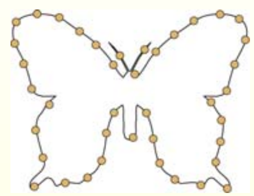


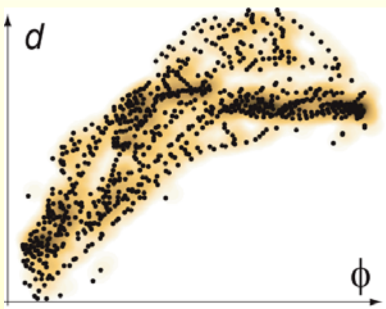
Reflection: Normal-Based Pruning



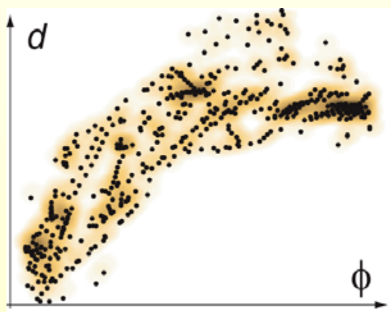


Point Pair Pruning

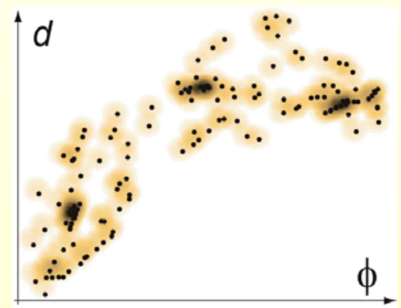




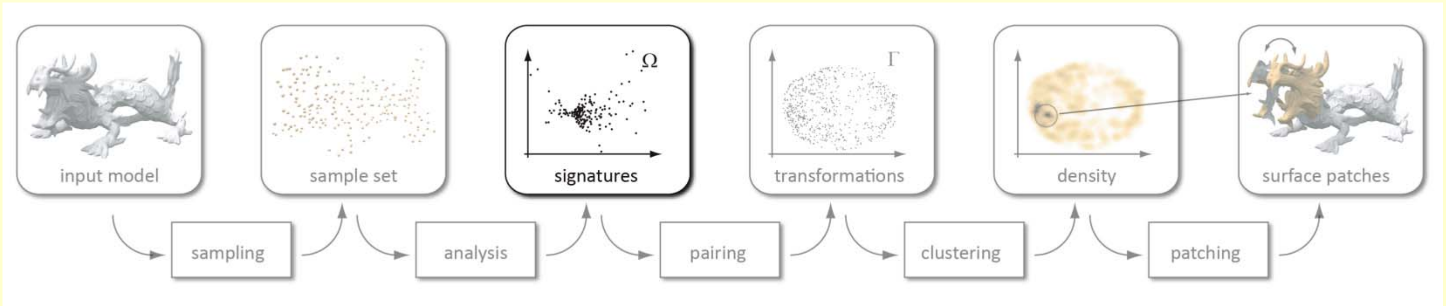
all pairs



curvature based



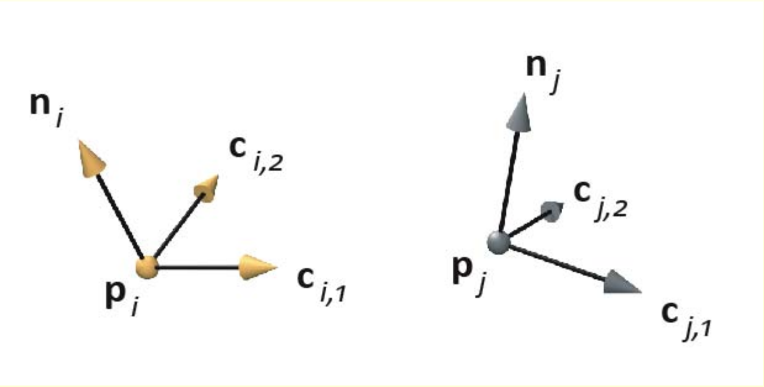
curvature + normal based

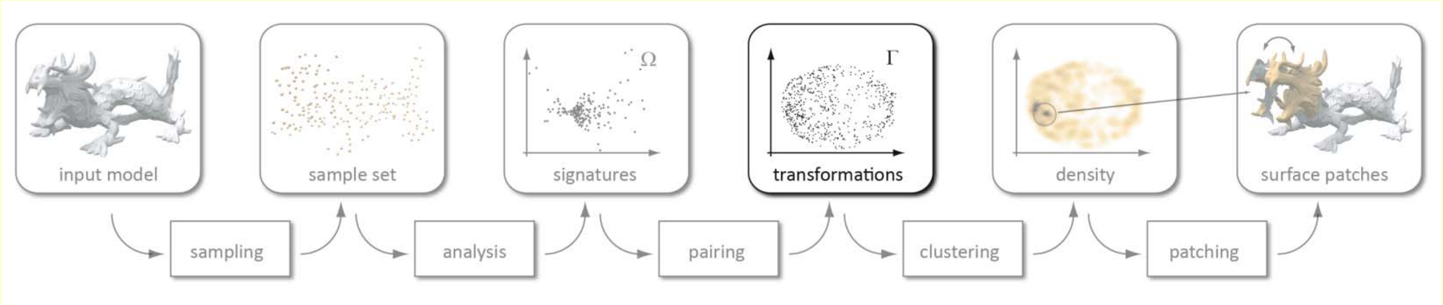


Transformations

- Reflection → point-pairs
- Rigid transform → more information

Robust estimation of principal curvature frames [Cohen-Steiner et al. `03]

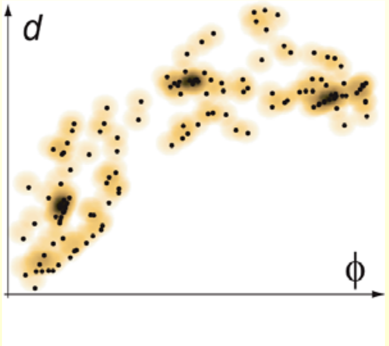


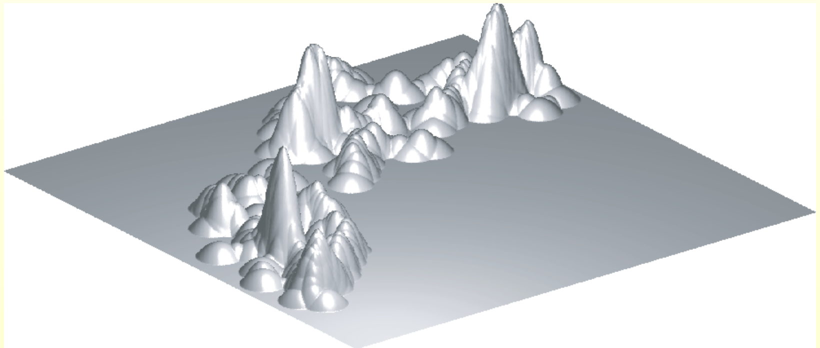


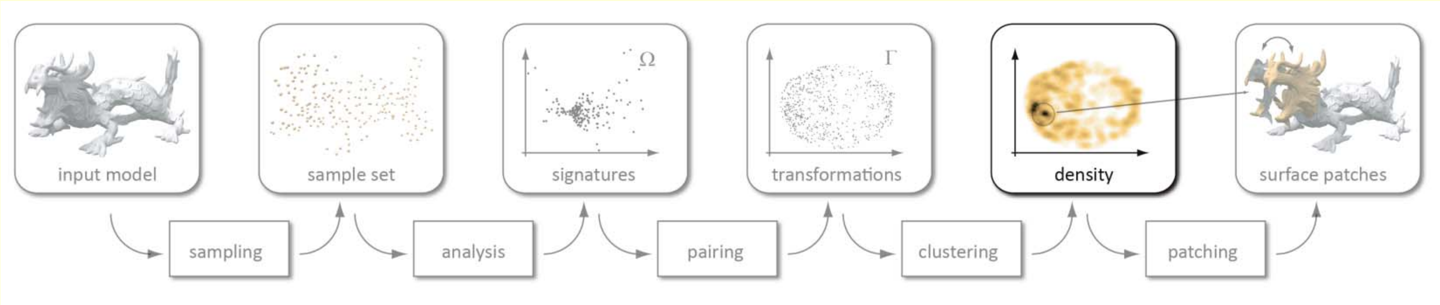
Mean-Shift Clustering

Kernel:

- Type $\rightarrow$ radially symmetric hat function
- Radius

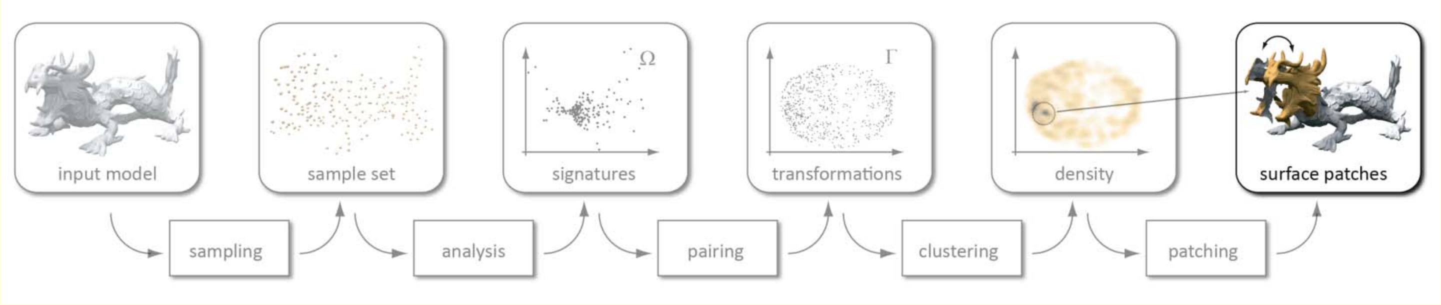






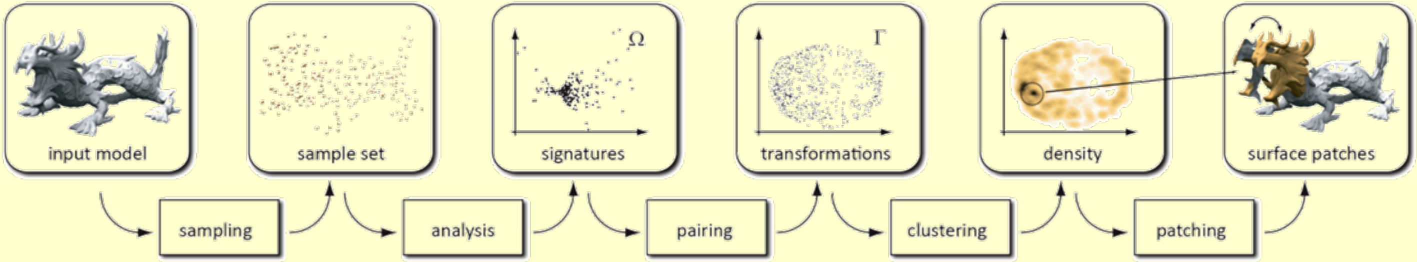
Verification

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



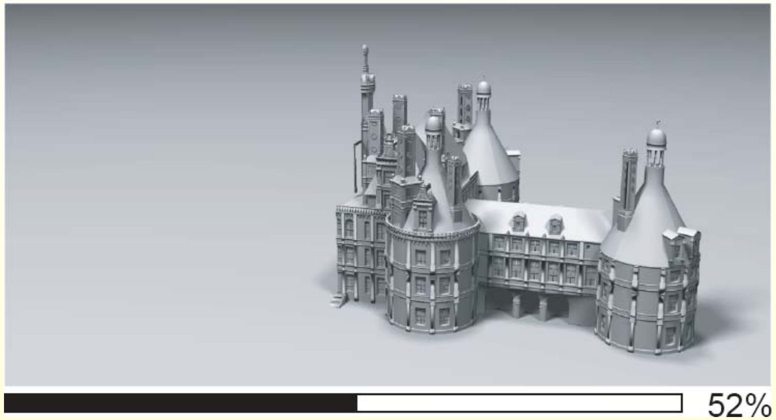
Random Sampling

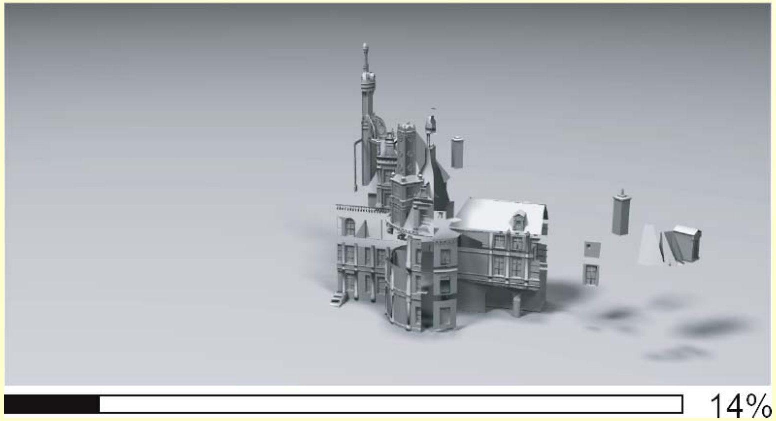
- Height of clusters related to symmetric region size
- Larger regions likely to be detected earlier
- Output-sensitive ...

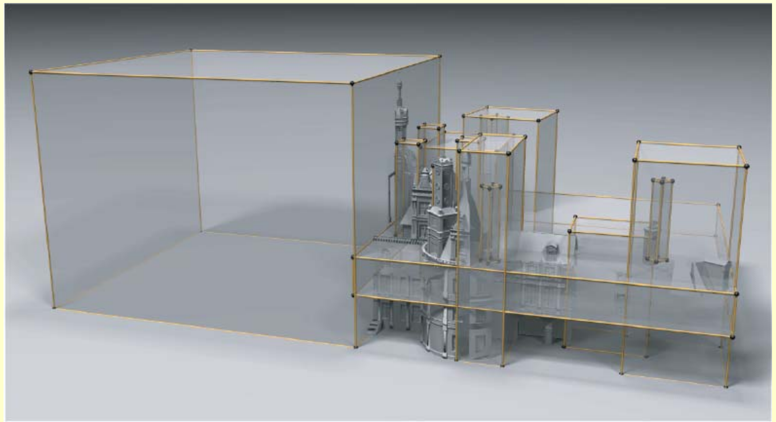


Compression: Chambord

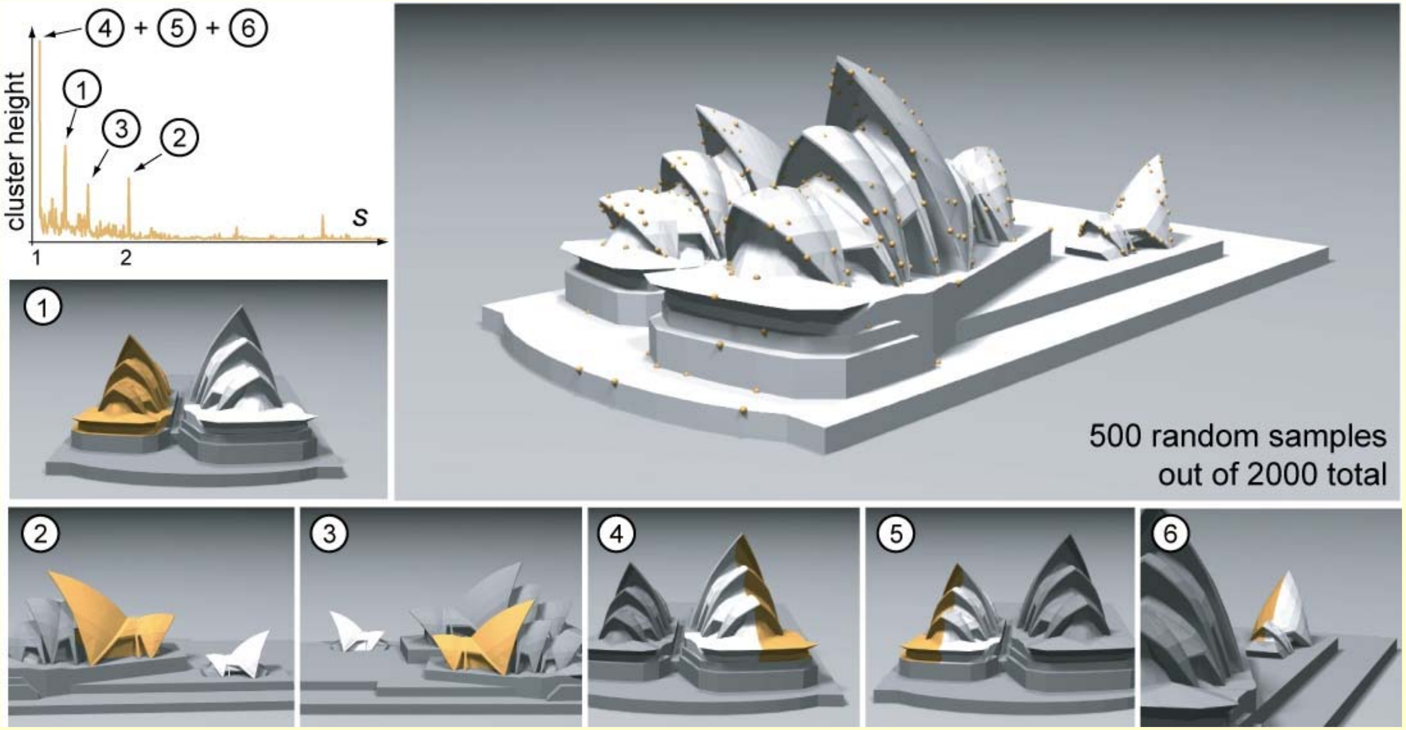
Compression: Chambord







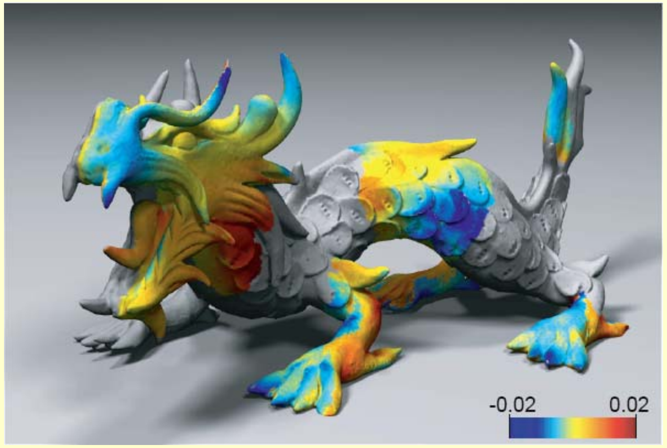
Opera



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

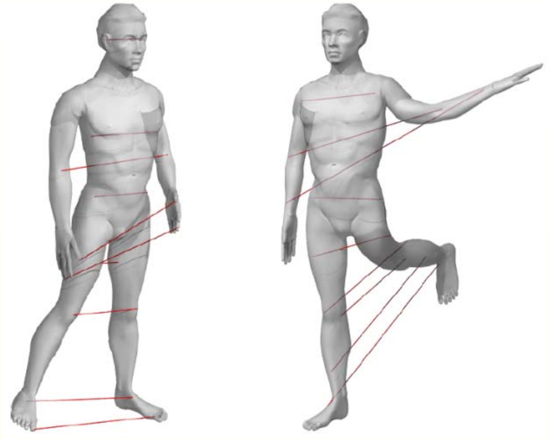
detected symmetries



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

Global Intrinsic Symmetries

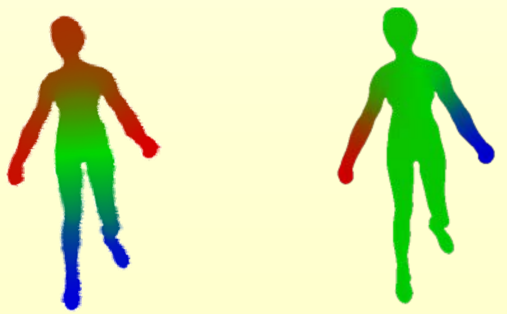
[Ovsjanikov, Sun, G., SGP 2008]

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

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

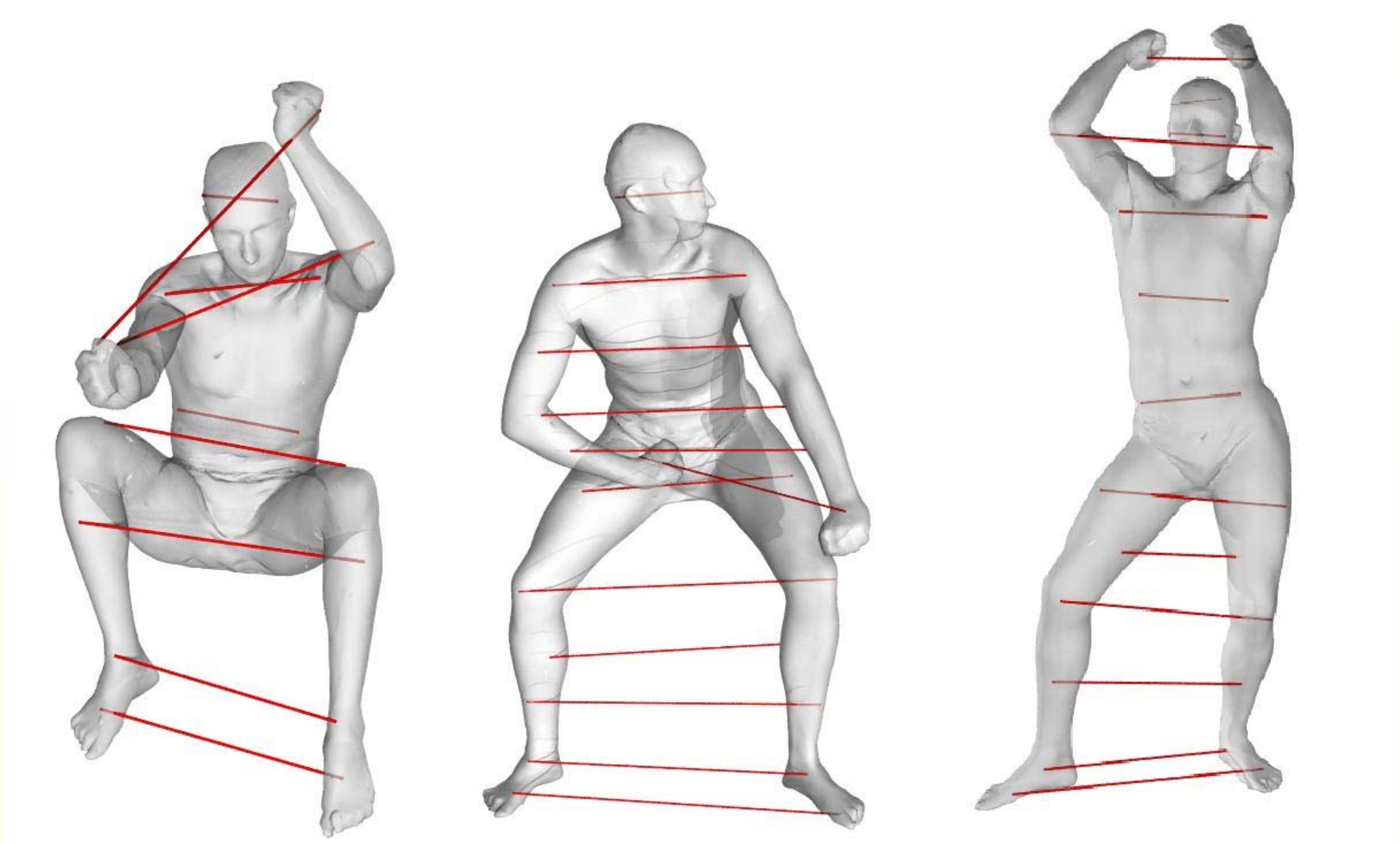
  - $\phi_i(\mathbf{p})$ is the value of the *i*-th eigenfunction of the Laplace-Beltrami operator at **p**
  - Invariant under isometric deformations
- Main Observation: Intrinsic symmetries of the object become extrinsic symmetries of the signature space.

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



Positive Negative

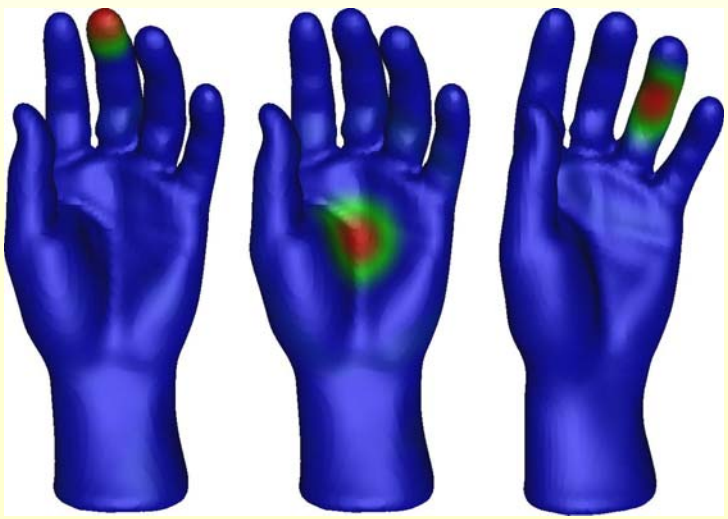
Global Intrinsic Symmetries

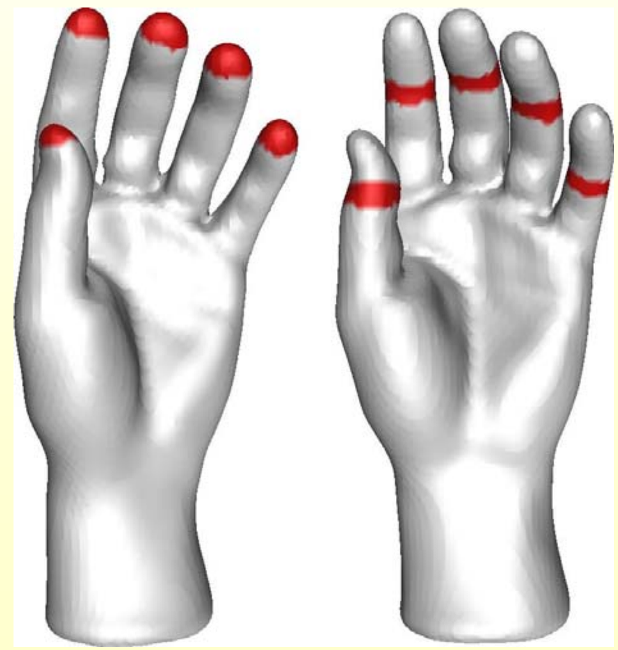


See MMDS poster on Friday

Partial Intrinsic Symmetries

- One part of an object is isometrically mapped to another part
- Use heat kernel
  - $k_t(x, y) = \sum_i e^{-\lambda_i t} \phi_i(x) \phi_i(y).$
  - Is the amount of heat transfered from y to x in t time.
  - $k_t(x, \cdot)$ is a bump function with scale t

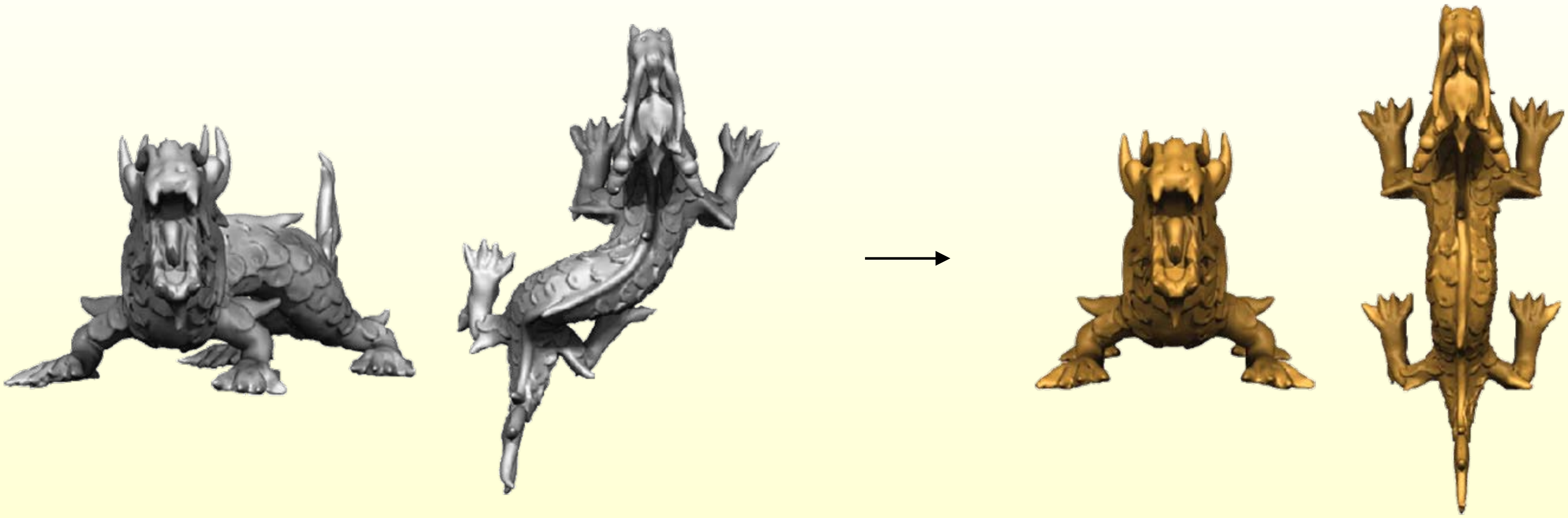




[Ovsjanikov, Sun, G., in prep. 2008]

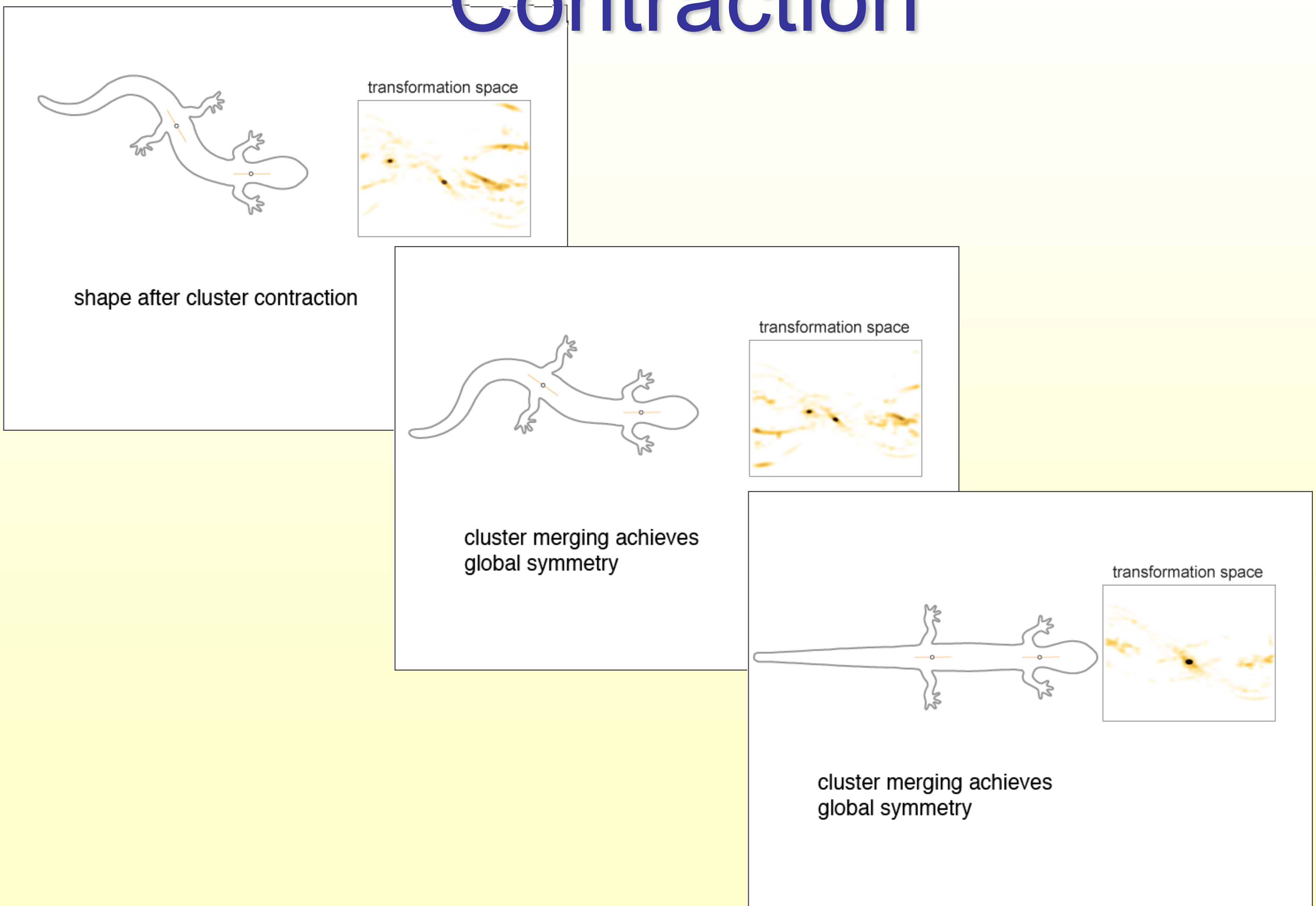
Extrinsic Symmetrization

Goal: Symmetrize 3D geometry



Approach: Minimally *deform* the model in the *spatial domain* by *optimizing* the distribution in *transformation space*

Cluster Enhancement and Contraction

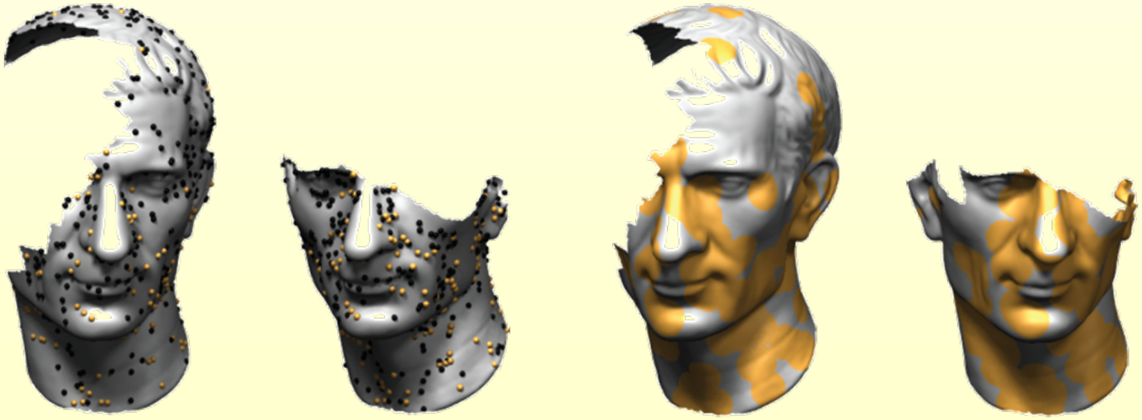


Key Points and Issues

- Capturing partial/approximate/intrinsic symmetries of 3D shapes can be done efficiently via a voting mechanism
- Only transforms supported by the data are searched and larger symmetries are found with less work

II. Distributed Congruence Discovery

[Pauly, Giesen, Mitra, G., SGP 2006]



Probabilistic Fingerprints

partial similarity

probabilistic fingerprint

Shapes are never directly compared

probabilistic fingerprint

compact

compare

independent

Insight

Partial matching $\rightarrow$ difficult problem

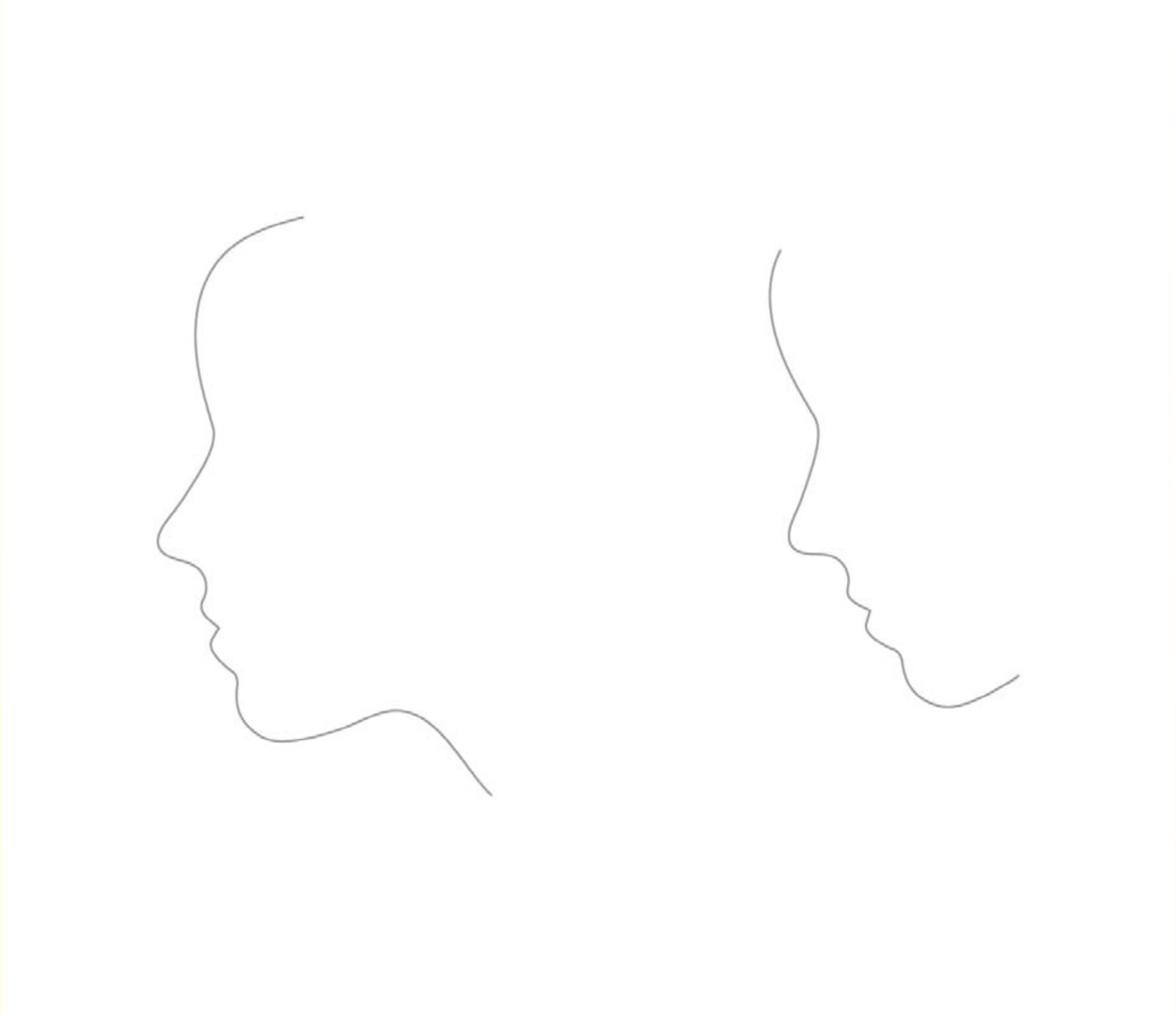
Total matching $\rightarrow$ easy problem

Reduce partial matching $\rightarrow$
many small total matching problems

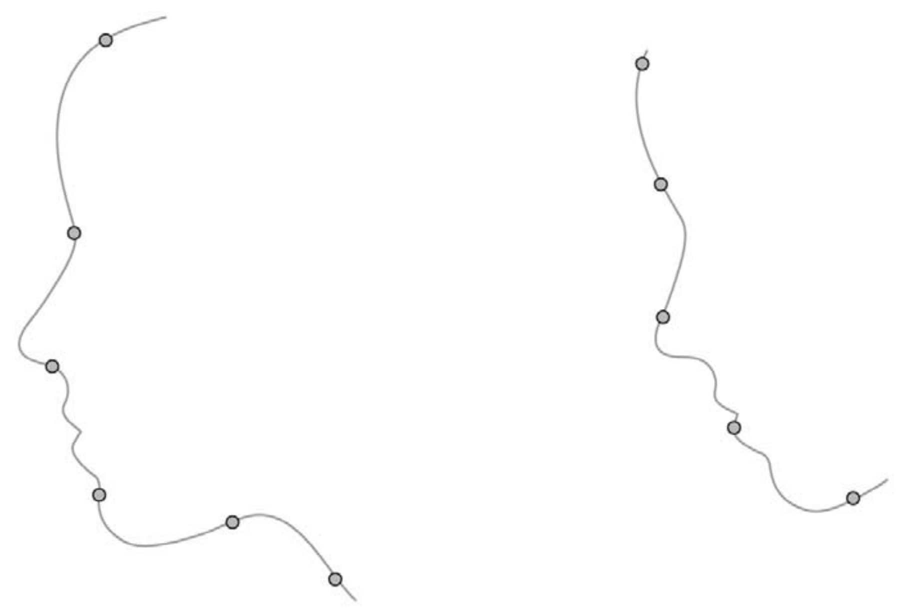
Results in few false positives $\rightarrow$
quick to verify and discard

From document similarity to shape similarity: shingles and min-hashing

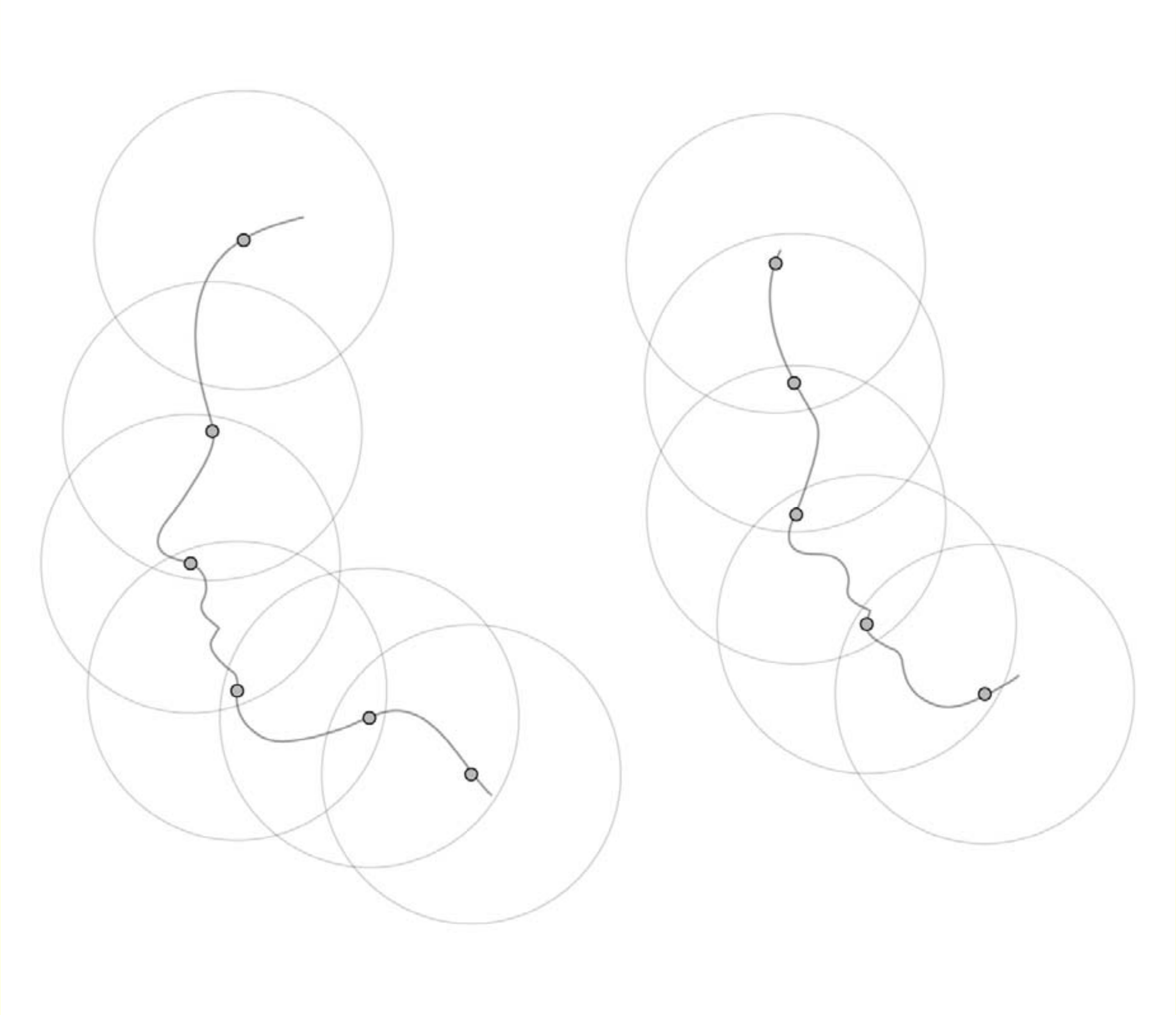
Input Shapes



Sample Points



Shingles: Overlapping Patches

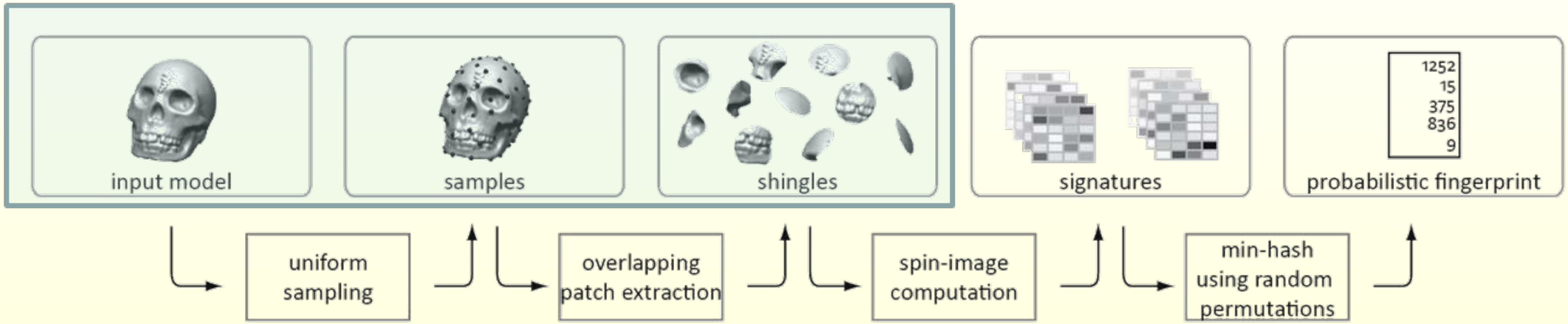


Shingles: Overlapping Patches

Bag of Patches: Ordering Discarded

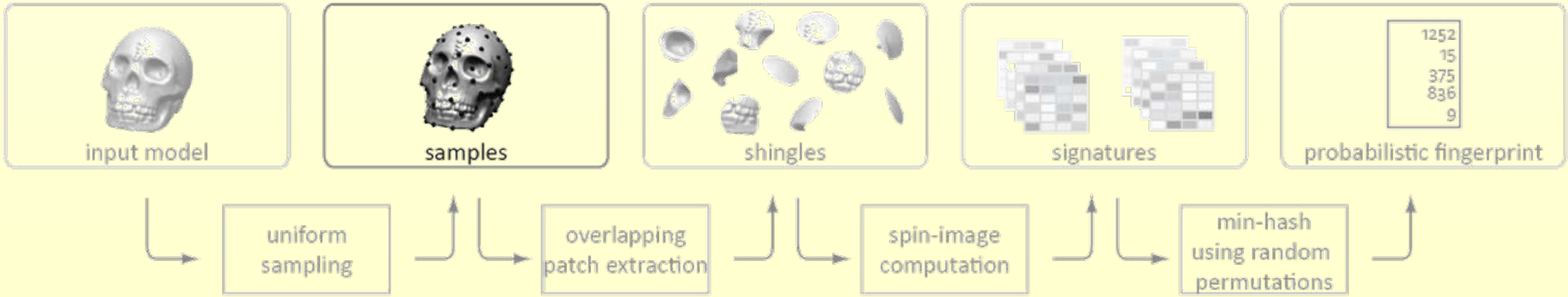
... but with sufficient ovelaps, can be recovered – DNA whole genome shotgun sequencing

Fingerprint Pipeline



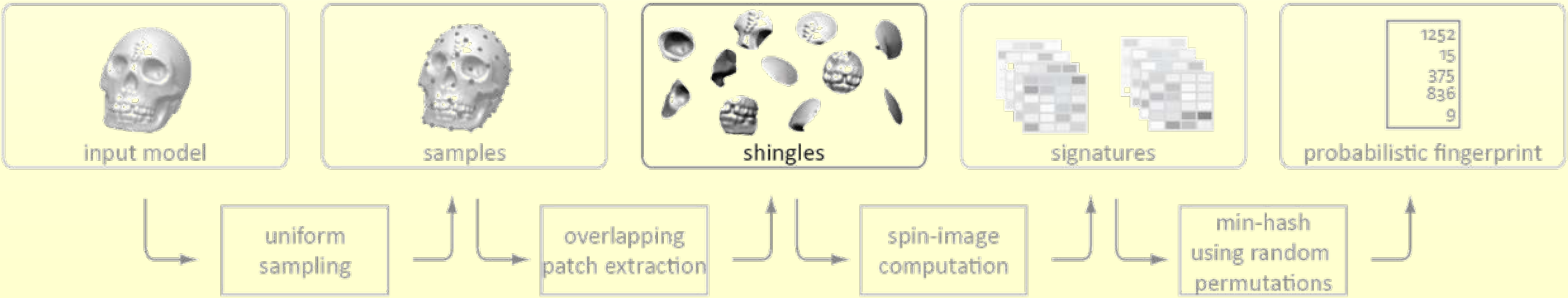
Pipeline: Uniform Sampling

- Uniform spacing → use [Turk`92]
- Sample spacing ≈ δ



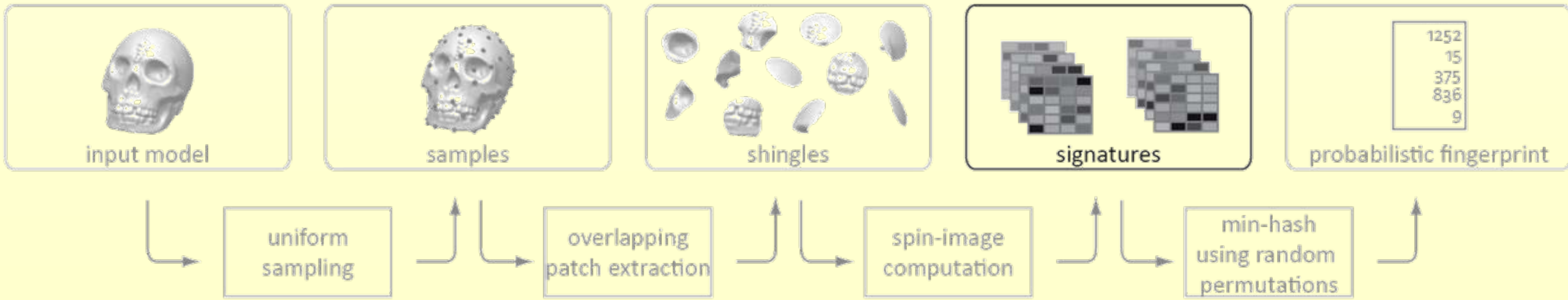
Pipeline: Shingle Generation

- Shingles: overlapping, unordered patches
- Shingle radius: ρ
- $\rho \gg \delta$



Pipeline: Signatures

- Stable signatures wrt. sampling (continuity)
- Invariant to rigid transforms
  - Spin-images [Johnson, Hebert 1999]
- Shape $\rightarrow$
  unordered high-dimensional point set with rigid transform factored out

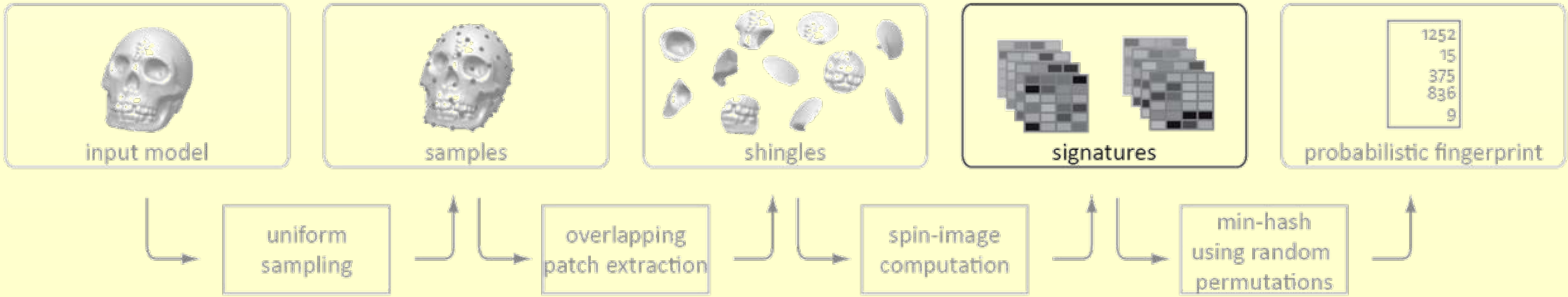


Pipeline: Resemblance

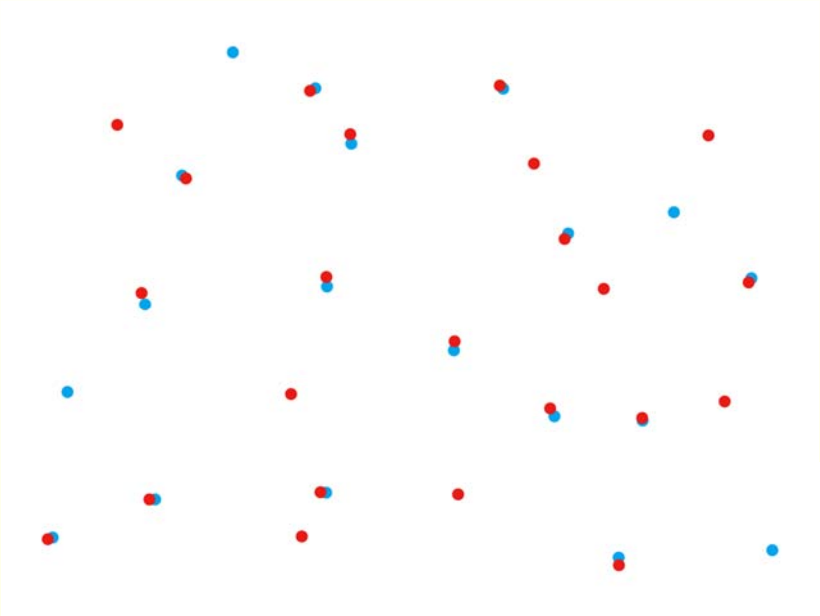
Jaccard similarity measure

$$r(S_1, S_2) = \frac{|\{s_1\} \cap \{s_2\}|}{|\{s_1\} \cup \{s_2\}|}$$

- Similarity/resemblance
  - Defined wrt. signatures
- Compare two bags of points in a high-d space
  - No alignment required
  - Still, brute force evaluation impractical



How to Compare Point Sets

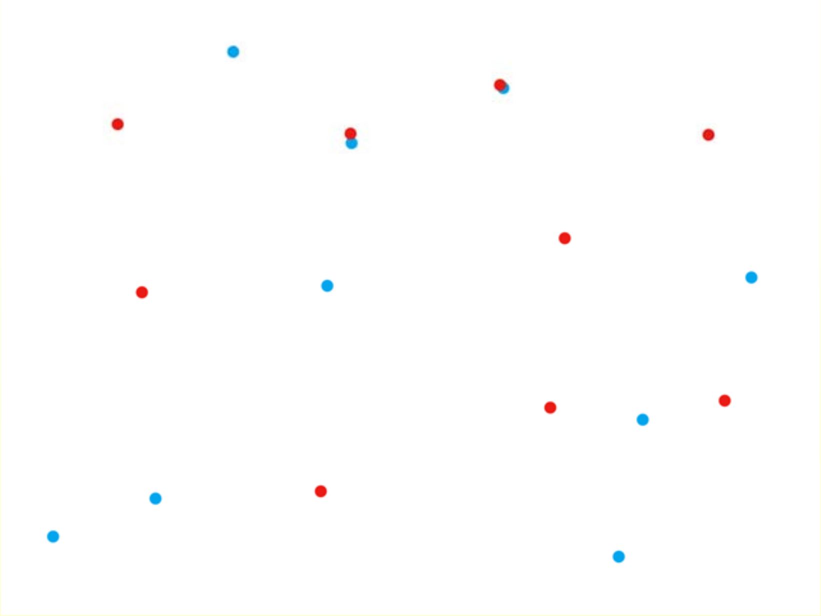


- Compare two point sets $\rightarrow$ no need to align

$$r(S_1, S_2) = \frac{|\{s_1\} \cap \{s_2\}|}{|\{s_1\} \cup \{s_2\}|}$$

- But, we don't have red and blue points together

Reduce Sample Size



We need consistent sampling

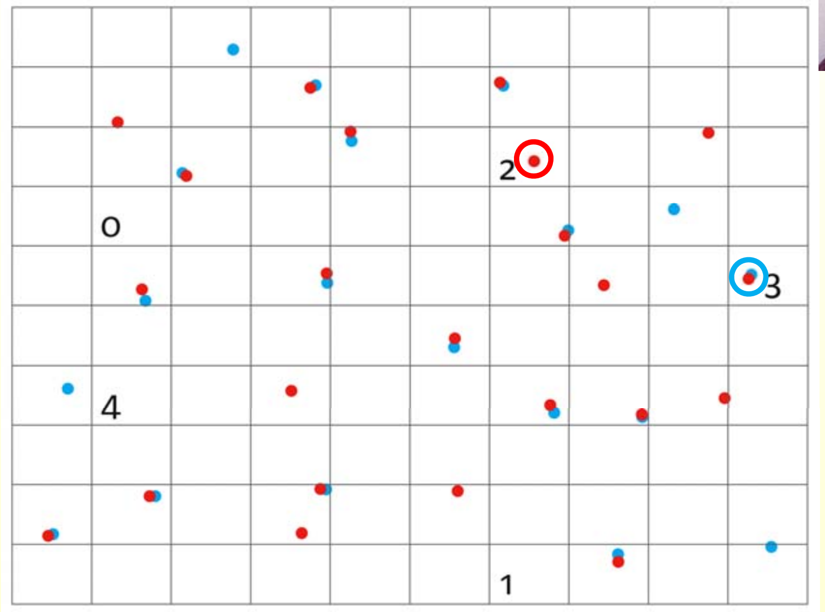
- Randomly sample red points
- Randomly sample blue points
- still need to solve for correspondences

} independently

Min-Hashing I: Using Random `Experts'

[Broder`97]

π_1 2 3

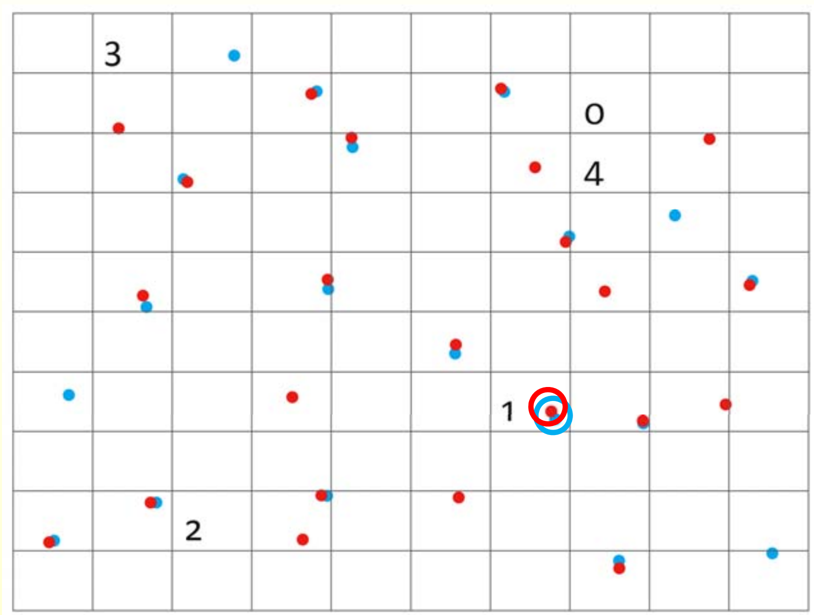


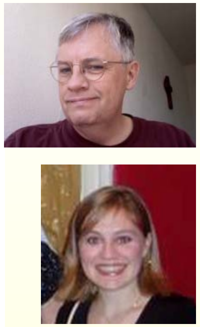
Each of *m* random 'experts'

- Has an ordering of space-boxes
- Selects the point that lies in lowest ordered box

$$min\{\pi(\mathcal{I})\}$$

Min-Hashing II





π_1 $\quad 2 \neq 3$

π_2 $\quad 1 = 1$

$$\Pr[f_i^1 = f_i^2] = R(S_1, S_2)$$

Each of m random ‘experts’

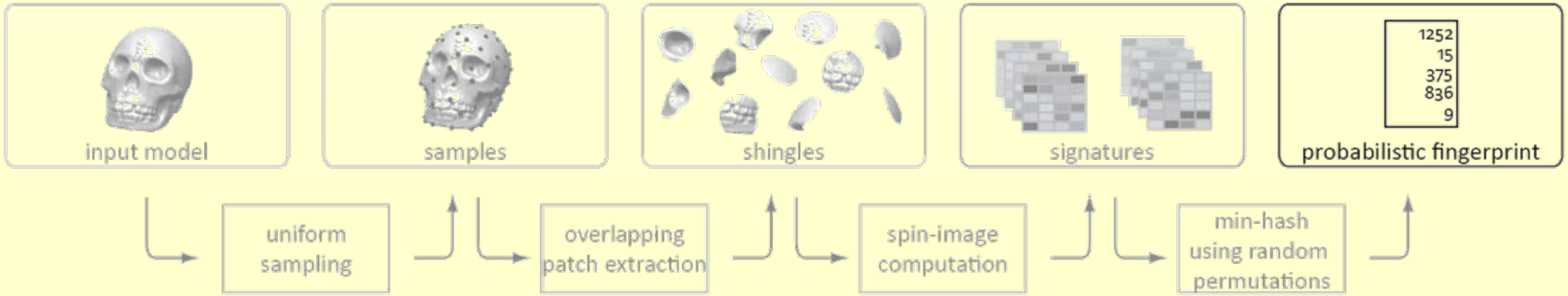
- Has an ordering of space-box
- Selects the point that lies in lowest ordered box

$$min\{\pi(\mathcal{I})\}$$

Pipeline: Min-Hashing

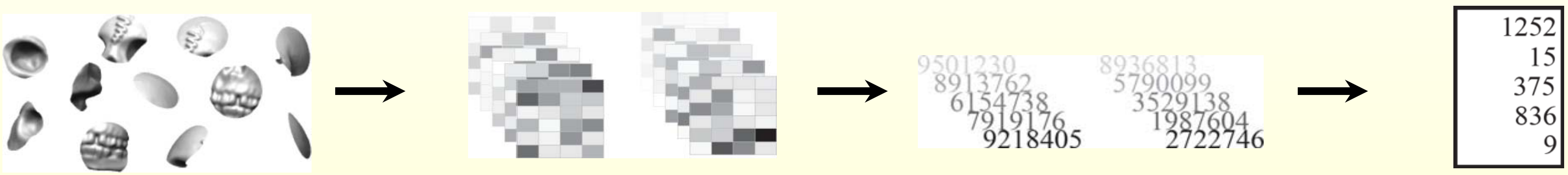
Feature selection by random experts

- ‘Features’ only useful for correspondence
  - Need not have any visual or semantic importance
- Reduces set comparison to element-wise comparison



Data Reduction

quantization | min hashing



Shingles → Signatures → Descriptors → Fingerprint

100k | 100k | 100k | 1k

set size remains constant | set reduction

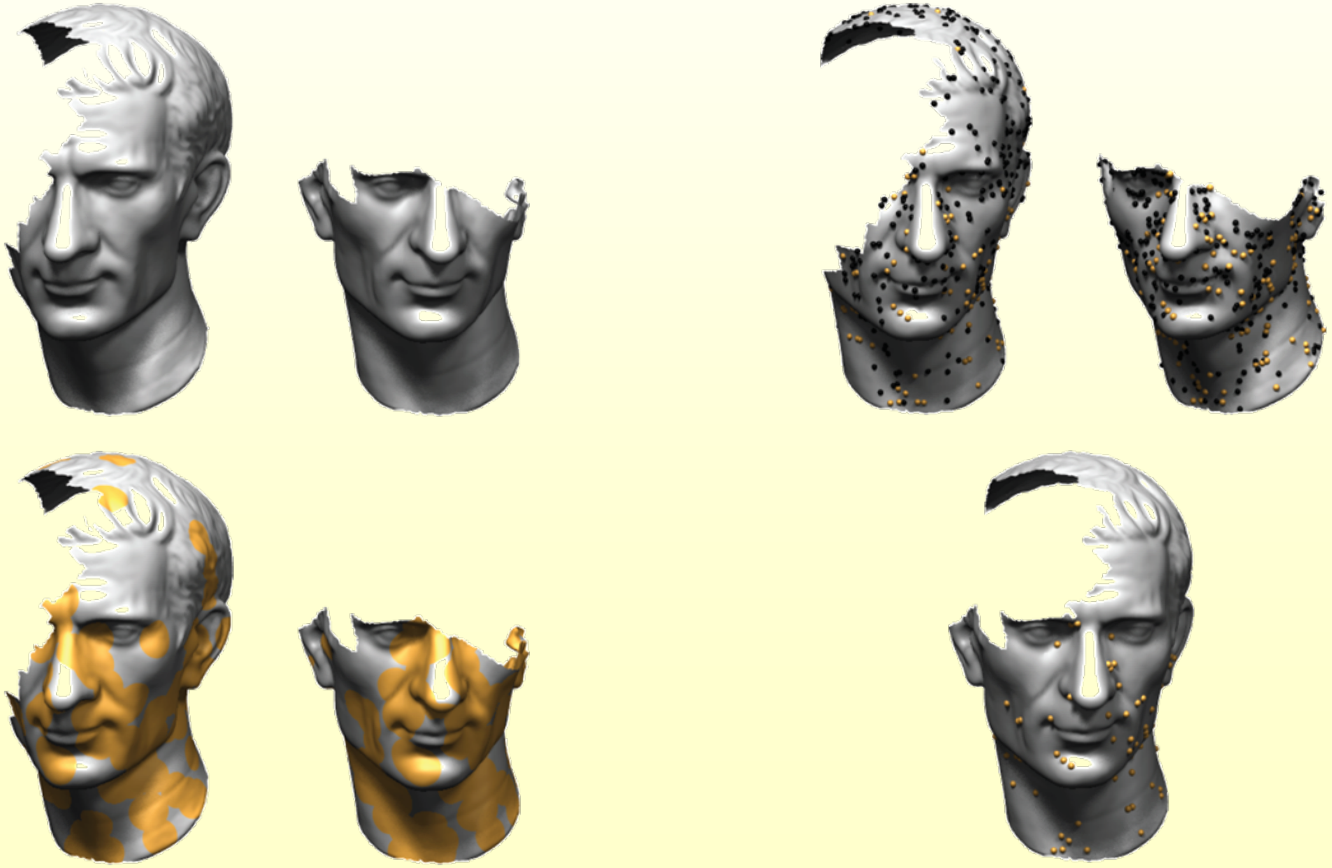
Applications

- 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	

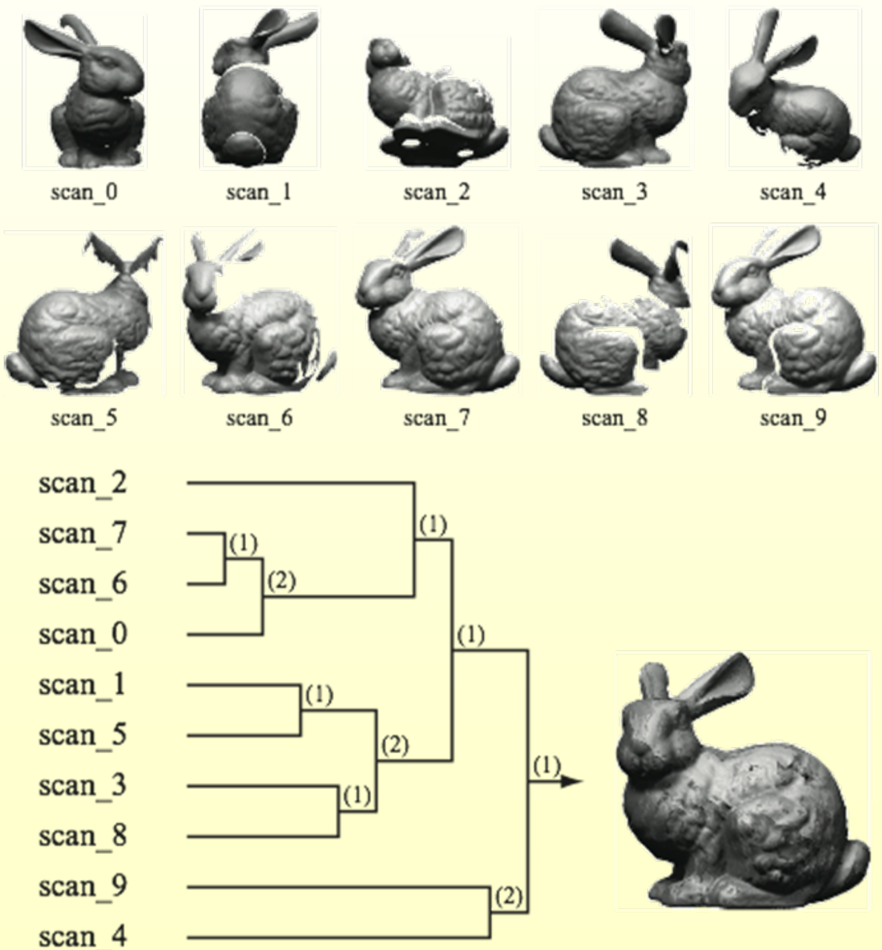
Applications

- Adaptive feature selection for stitching



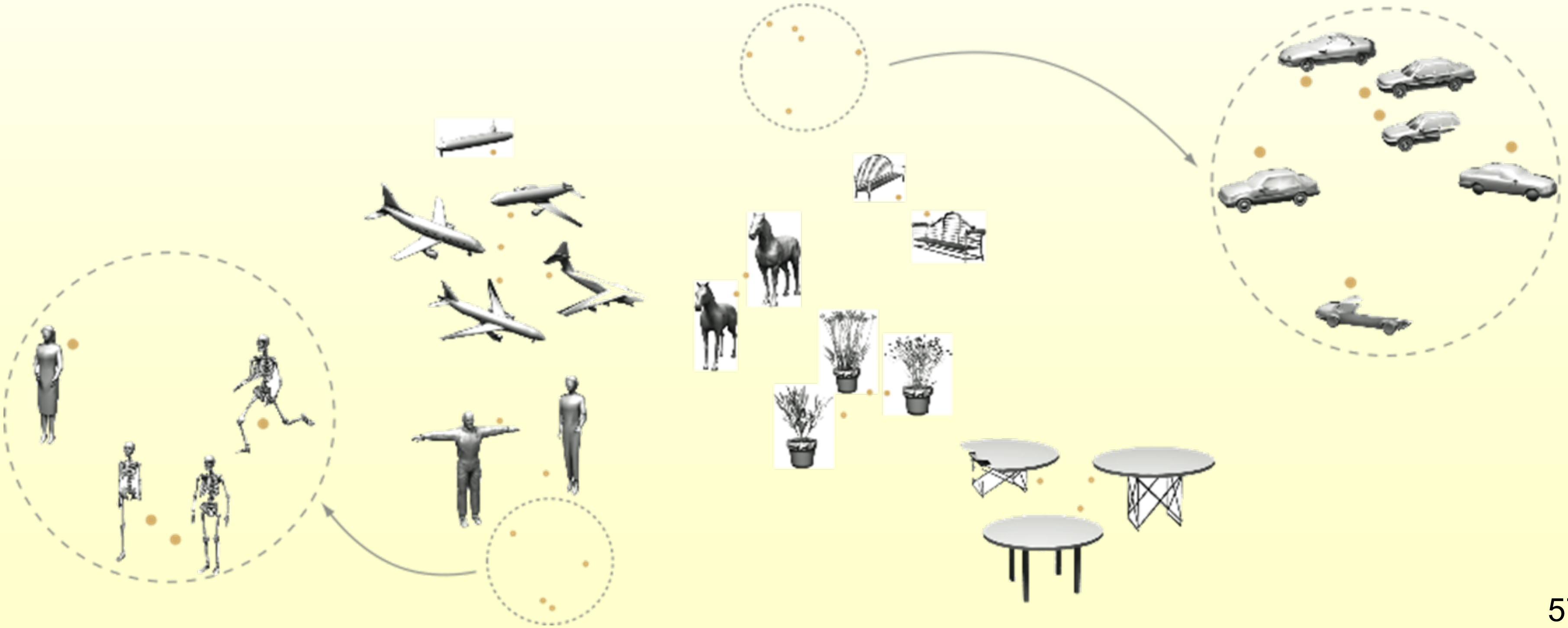
Applications

- Multiple scans
    - greedy alignment using priority queue
    - fingerprint matching determines score
    - advanced alignment method for verification
    - merging fingerprints requires no re-computation



Applications

- Shape distributions



Key Points and Issues

- Resemblance defined as set operation on signature sets → quantization is crucial
- Random experts effectively extract consistent set of features → requiring no explicit correspondences
- Fingerprints do not preserve spatial relation of shingles → false positives are possible
- Few parameters that are easy to tune

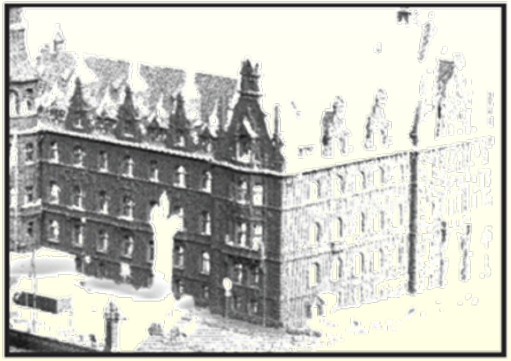
III. Repeated Pattern Detection

[Pauly, Mitra, Wallner. G., and Pottmann, Siggraph ’08]

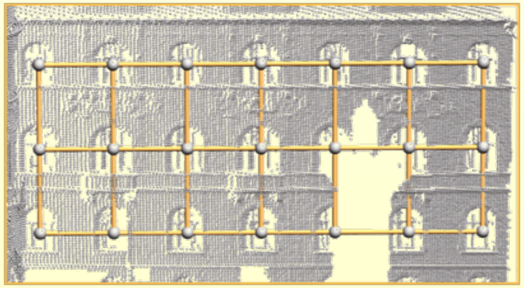
Structure Discovery

- 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



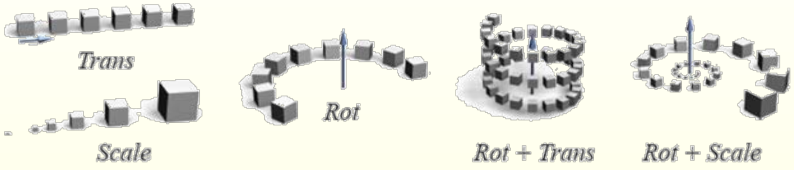
Input Model



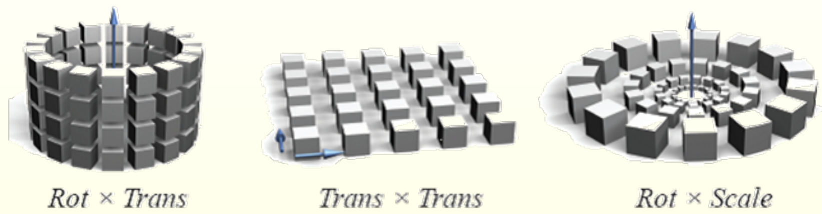
Regular structure

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

Algorithm Overview



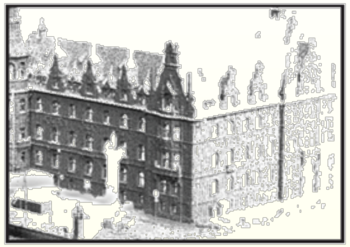
1D structures



2D structures

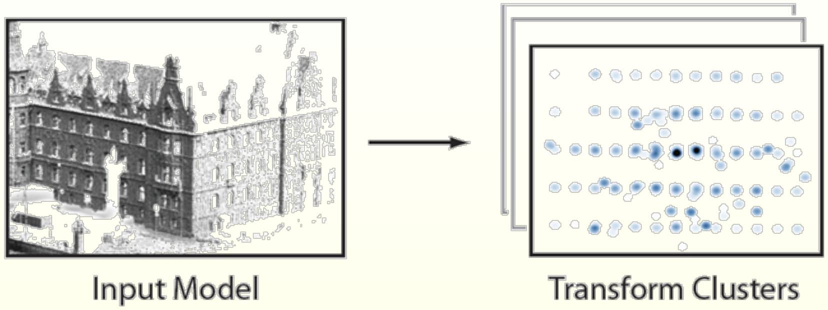
Regular structures:
rotation + translation + scaling → any commutative combinations in the form of 1D, 2D grid structures

Algorithm Overview



Input Model

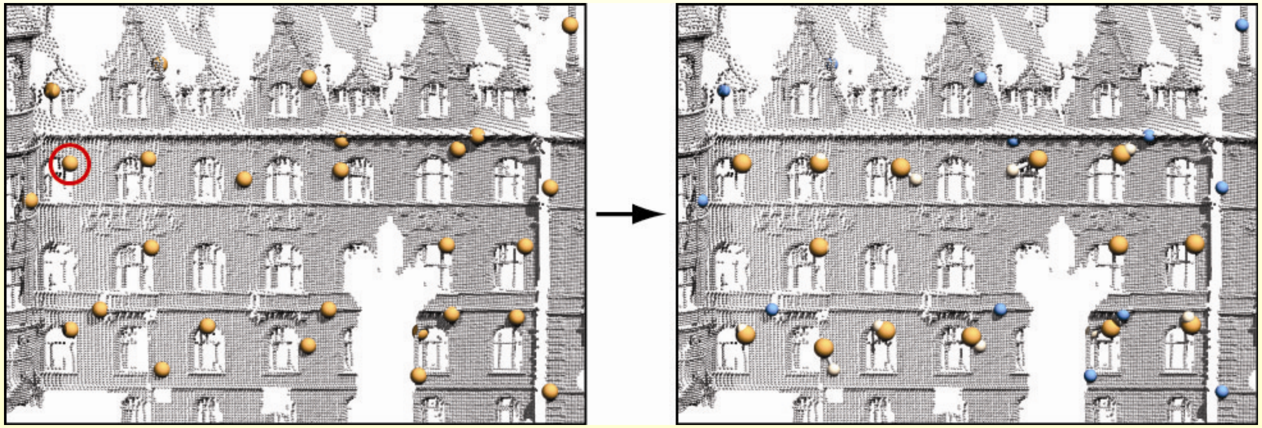
Algorithm Overview



- Transform Analysis
  - map to suitable transform space
  - goal: enhance and amplify regularity signal

Similarity Sets

Compare all pairs of small patches, using local shape descriptors

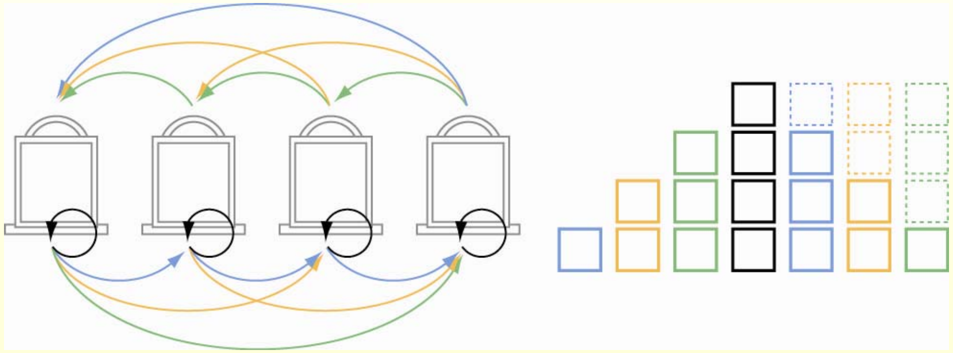


Based on shape descriptors alone

Pruned, after validation w. geometric alignment

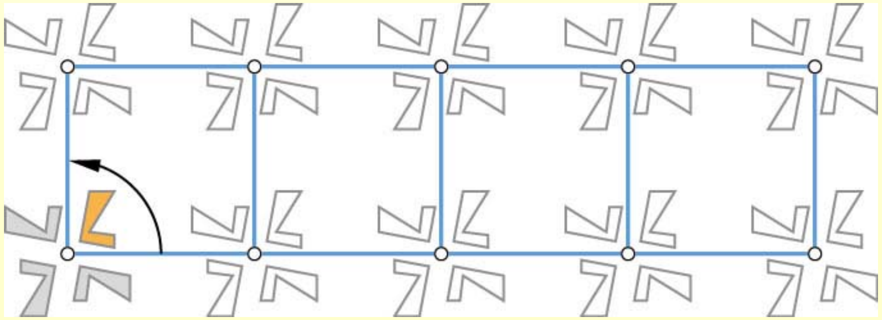
Transform Analysis

- Commutative 1- and 2-parameter groups

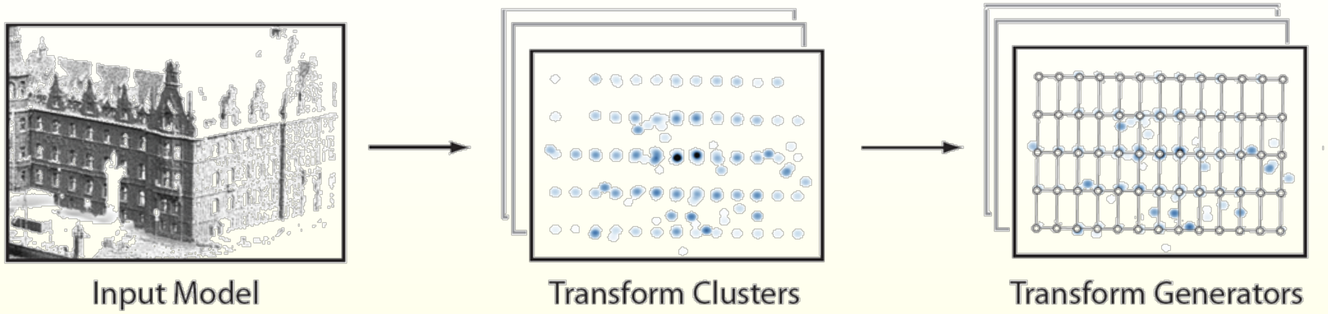


Match small local patches of geometry

Patterns in 3D space
map to patterns in transform space

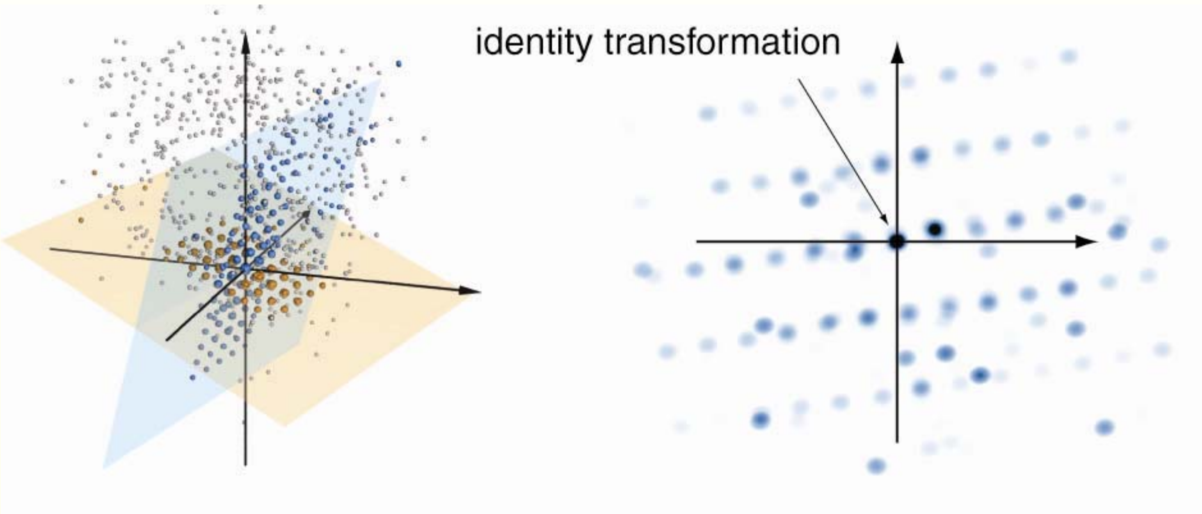


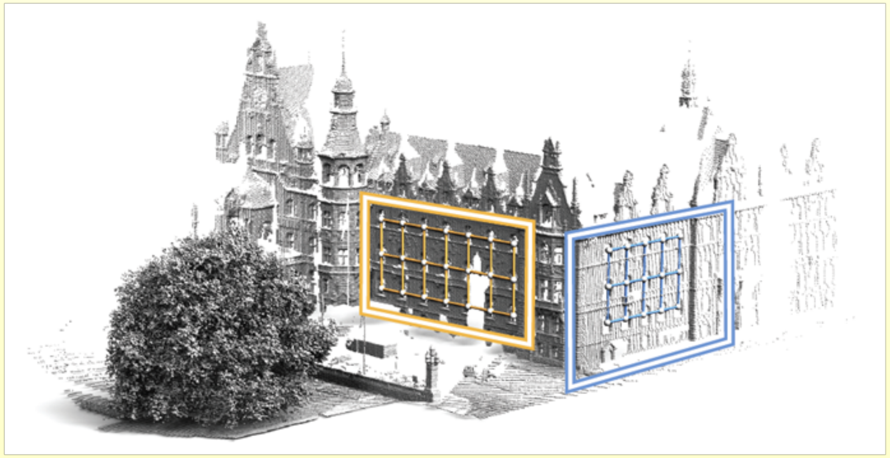
Algorithm Overview



- Transform Analysis
  - map to suitable transform space
- Model Estimation
  - under a suitable parametrization, all previous patterns correspond to 1- or 2-d grids
  - robust grid estimation with noisy/partial data in transform space

Model Estimation





Grid Fitting I

Clusters in transform space

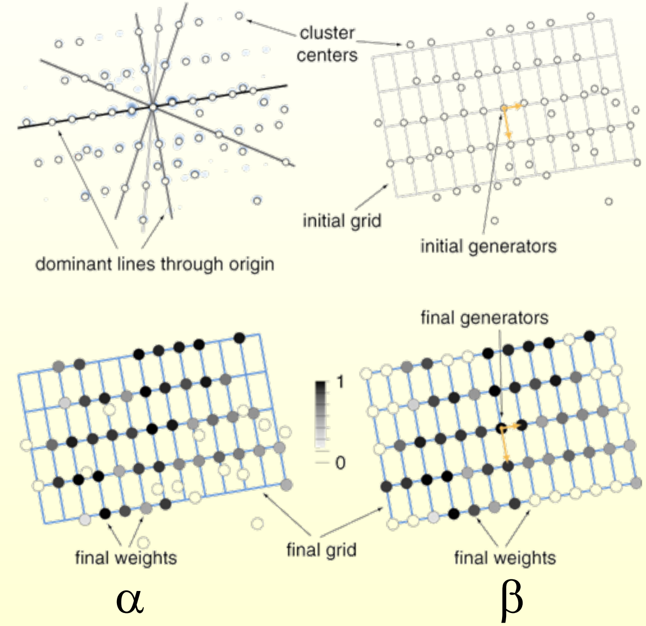
$$\vec{g}_1, \vec{g}_2, \{\alpha_{ij}\}, \{\beta_k\} = \underset{\vec{g}_1, \vec{g}_2, \{\alpha_{ij}\}, \{\beta_k\}}{\operatorname{argmin}} E$$

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

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

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

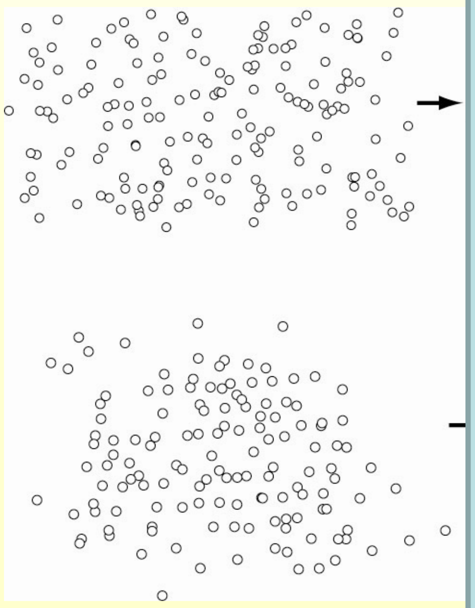
$$E_\alpha = \sum_i \sum_j (1 - \alpha_{ij}^2)^2 \qquad E_\beta = \sum_k (1 - \beta_k^2)^2$$



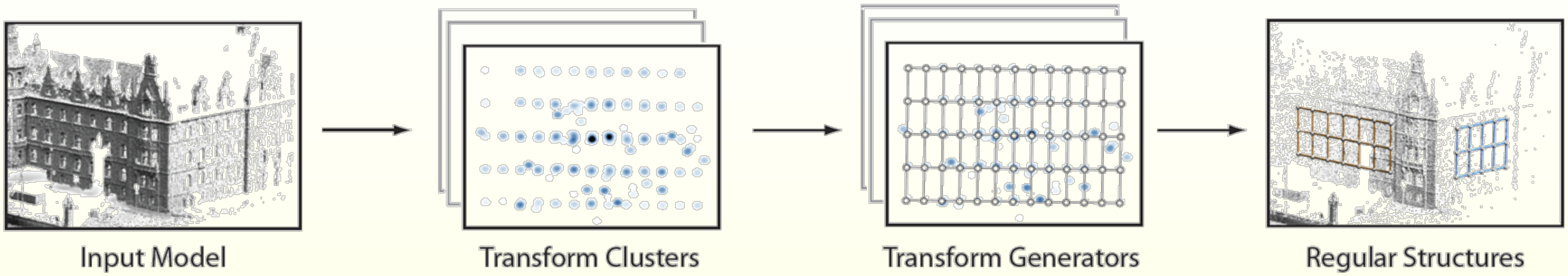
X = grid
C = transform cluster

Grid Fitting II

Finding grids amidst clutter



Algorithm Overview



- Transform Analysis
  - map to suitable transform space
- Model Estimation
  - robust grid estimation with noisy/partial data
- Aggregation
  - simultaneous optimization of regular structure + patch

Aggregation

- Once the basic repeated pattern is determined, we simultaneously (re-)optimize the pattern generators and the repeating geometric element it represents, by 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

The Math

We optimize a generating transform T represented by 4x4 matrix *H*, by trying to improve the alignment of all patches put into correspondence by *T*, using standard ICP techniques

$$\vec{H}_+ \approx \vec{H} + \epsilon \vec{D} \cdot \vec{H},$$

$$\vec{D} = \begin{pmatrix} \delta & -d_3 & d_2 & \bar{d}_1 \\ d_3 & \delta & -d_1 & \bar{d}_2 \\ -d_2 & d_1 & \delta & \bar{d}_3 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$T_+(\vec{x}) \approx T(\vec{x}) + \epsilon\left(\vec{d} \times T(\vec{x}) + \delta T(\vec{x}) + \vec{\bar{d}}\right)$$

$$T_+^k \approx (\vec{H} + \epsilon \vec{D} \cdot \vec{H})^k \;\rightarrow\; \vec{H}_+^k \approx \vec{H}^k + \epsilon f_k(\vec{H}, \vec{D}) + \epsilon^2(\ldots), \quad \text{with}$$

$$f_k(\vec{H}, \vec{D}) = \vec{D} \cdot \vec{H}^k + \vec{H} \cdot \vec{D} \cdot \vec{H}^{k-1} + \cdots + \vec{H}^{k-1} \cdot \vec{D} \cdot \vec{H}$$

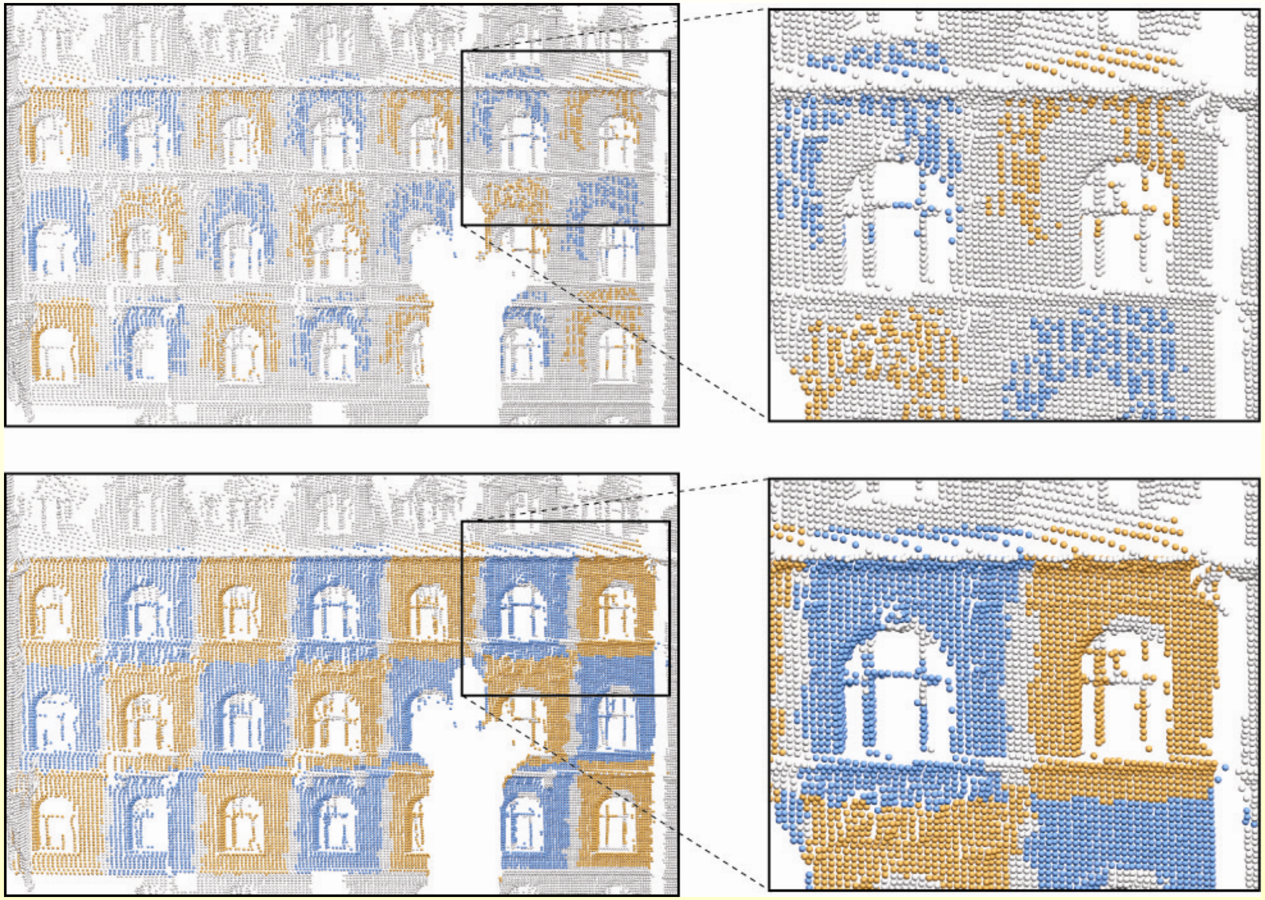
$$Q_{ij} := \sum\nolimits_l \left([(T_+^k(\vec{x}_l) - \vec{y}_l) \cdot \vec{n}_l]^2 + \mu [T_+^k(\vec{x}_l) - \vec{y}_l]^2 \right)$$

$$F(\epsilon \vec{D}) = \sum\nolimits_{i,j} Q_{ij}$$

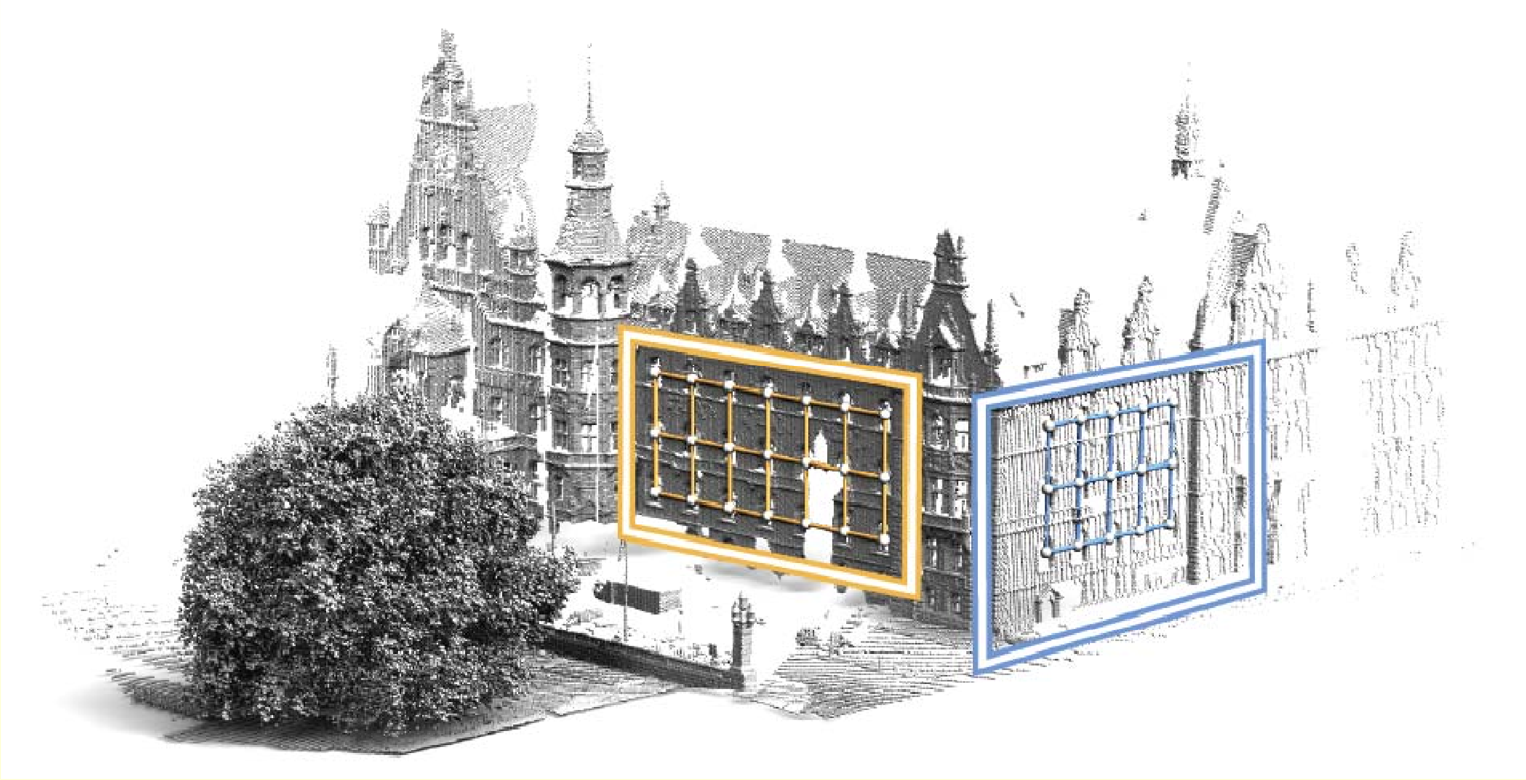
Simultaneous Registration

From grid optimization

After aggregation



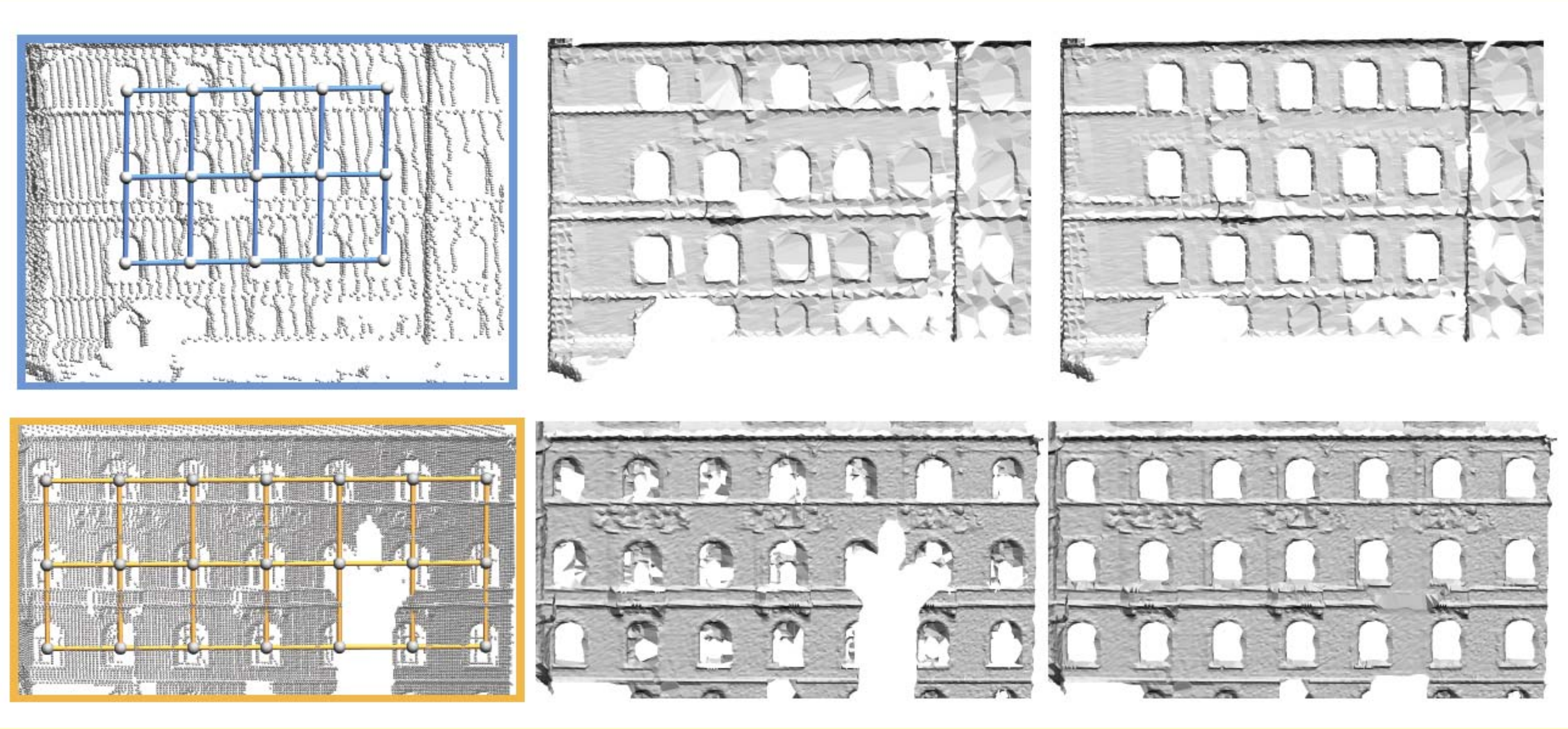
Scanned Building Facade



Output:

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

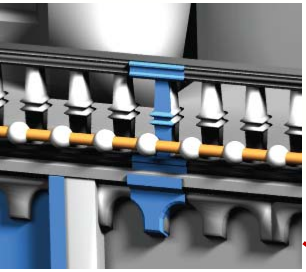
(Structural) Model Completion

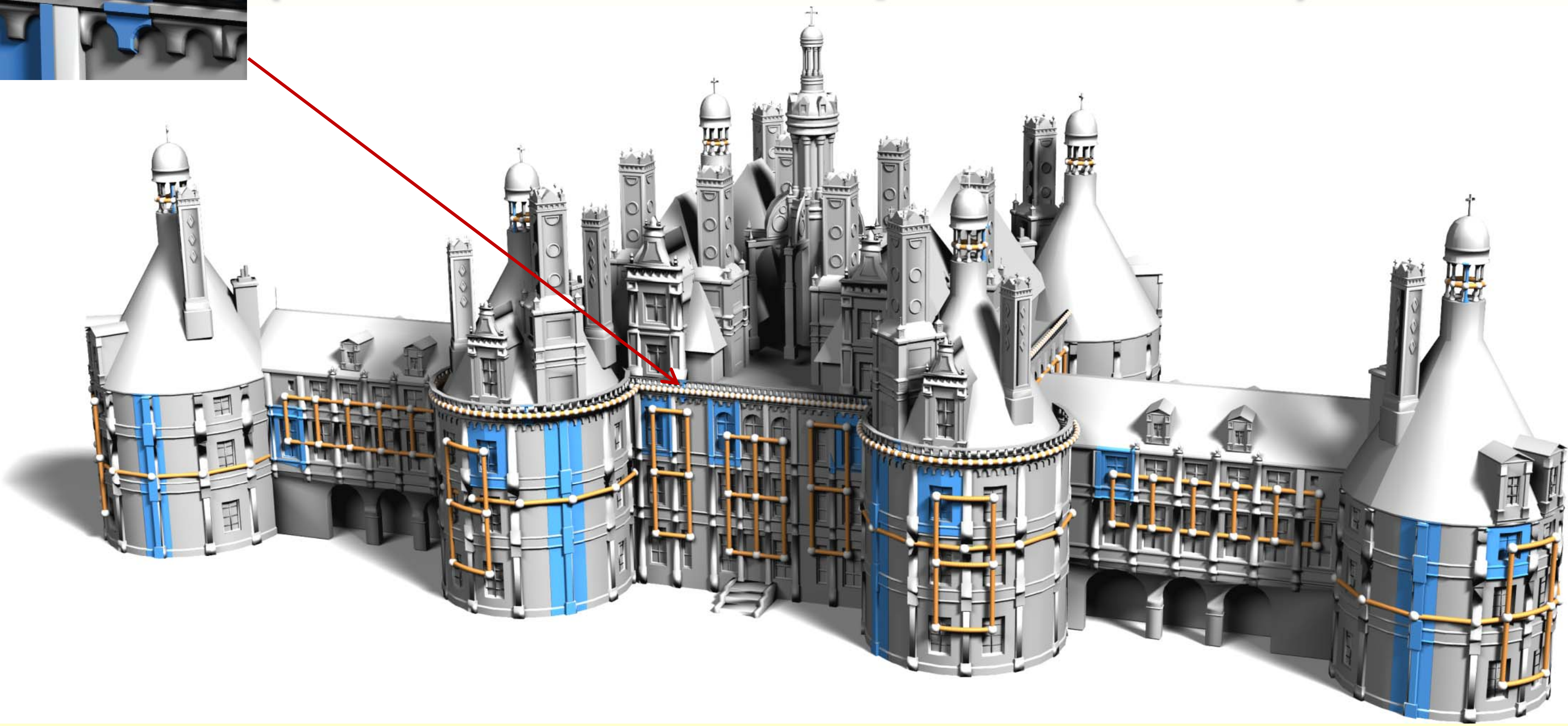


Naïve reconstruction

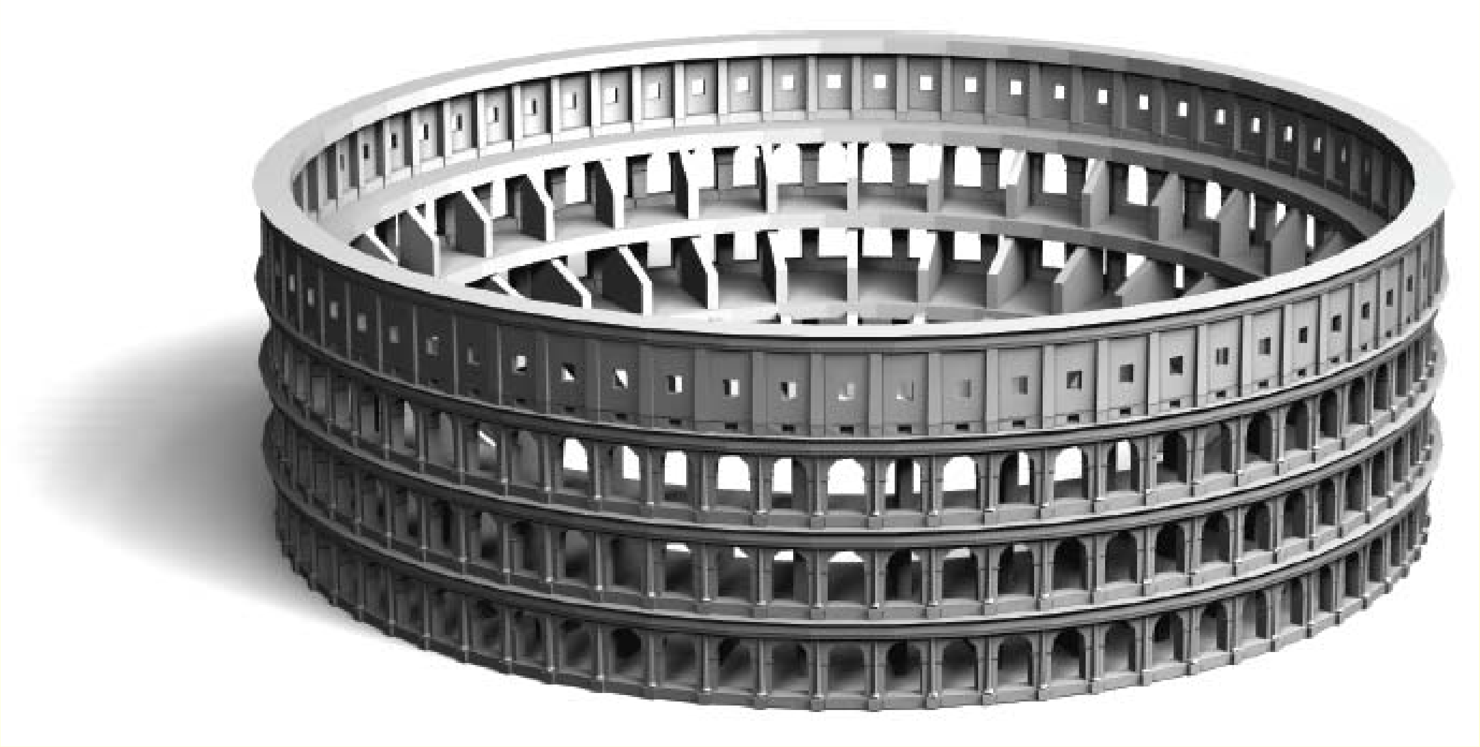
Reconstruction with structural constrains

Back to Chambord (30-100K Sample Points)

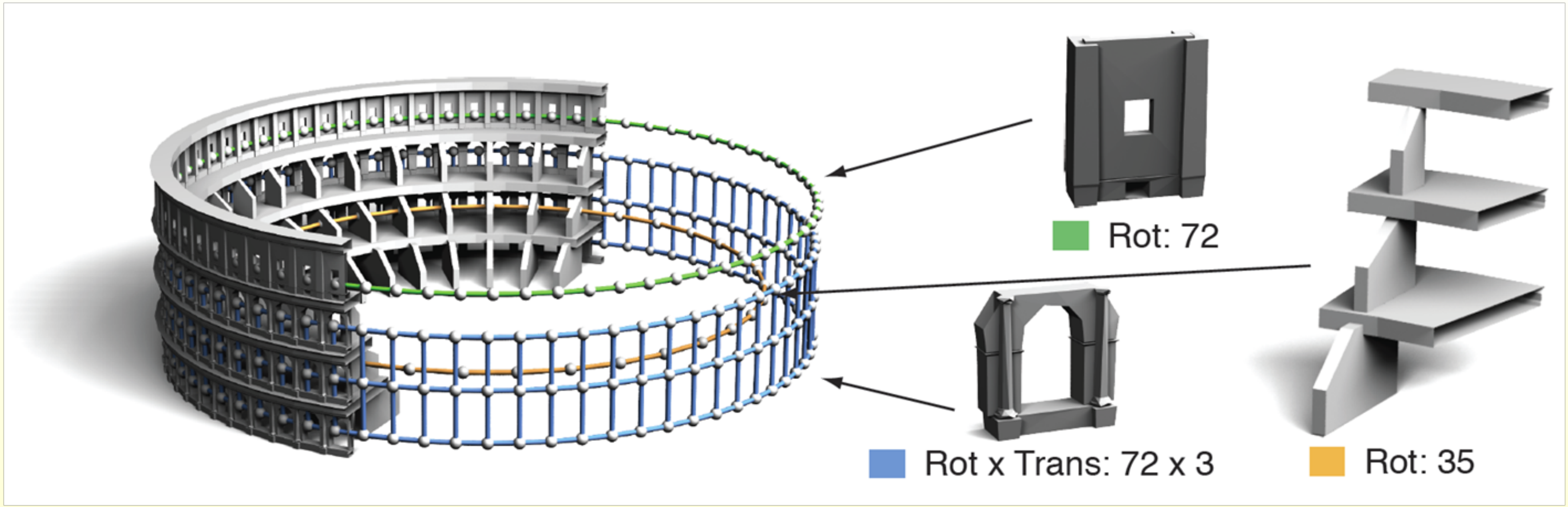




Amphitheater

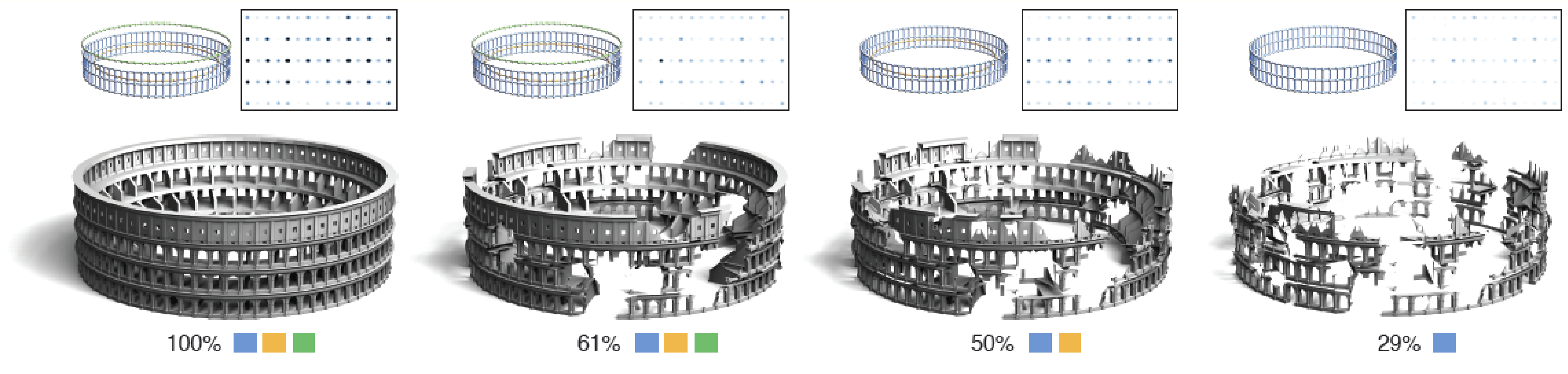


Amphitheater



Output: 3 grids + associated patches

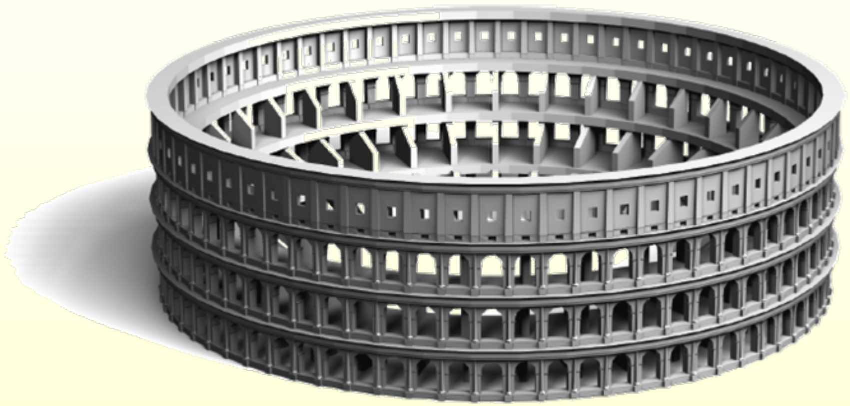
Robustness to Missing Data



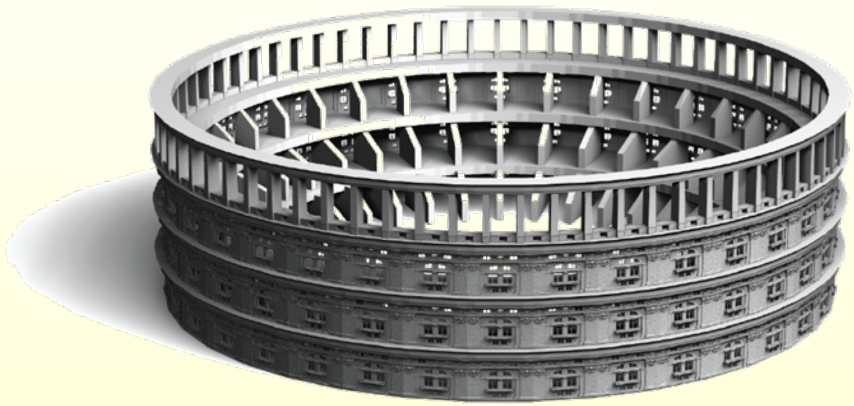
- More regular the structure → more resilient to missing data.

- Top row shows the corresponding grids in transform space plots.

Structural + Geometric Edit



Original

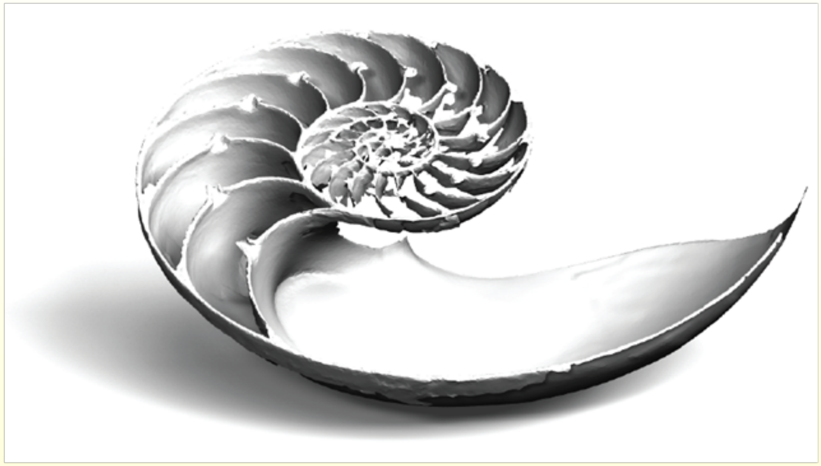


Edited

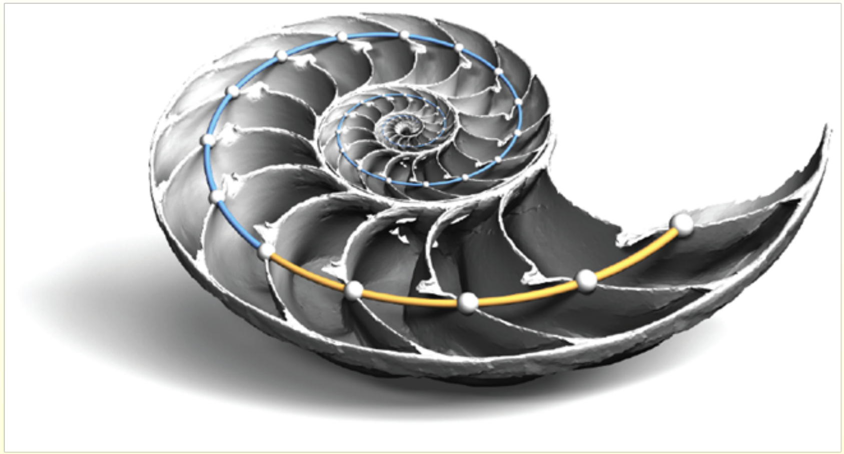
Nautilus: Similarity Transform

Input: 72 registered laser scans

Nautilus: Similarity Transform



Original



Edited

Output: Detected structure + growth

Key Points and Issues

- Patterns in 3D data map into patterns in the space of locally aligning transforms
- Grid fitting w. weights as optimization variables allows for missing data and outliers
- The full geometry is exploited in detecting the optimal repeating element and pattern generator(s)
- Related to non-local smoothing in images

From 3-D to Any-D

- Presented work on structure extraction for 3-D data sets of scanned geometry
- 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?

Acknowledgements

- Collaborators:
  - Current and past students: Niloy Mitra, Maks Ovsjanikov
  - Current and past postdocs: Jian Sun, Mark Pauly
  - Senior: Joachim Giesen, Mark Pauly, Helmut Pottmann, Johannes Wallner
- Sponsors:

U.S. DEPARTMENT OF HEALTH AND HUMAN SERVICES
National Institutes of Health

National Science Foundation
WHERE DISCOVERIES BEGIN

DARPA