Digital Geometry - Shape Matching

Junjie Cao @ DLUT

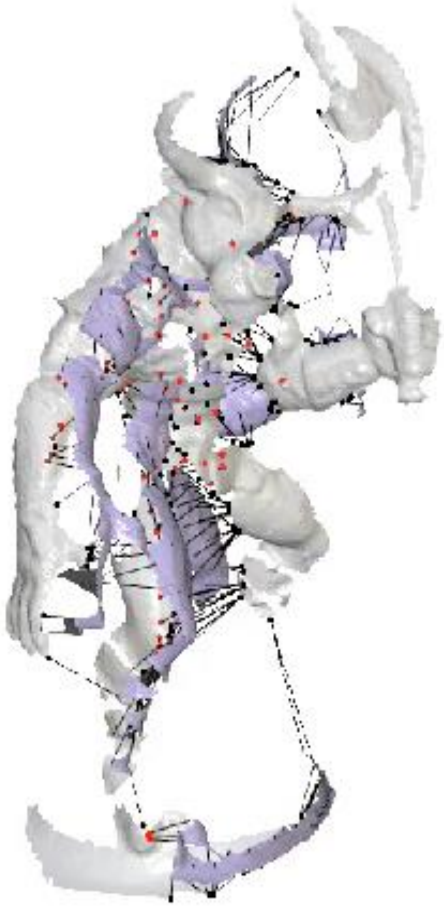
Spring 2018

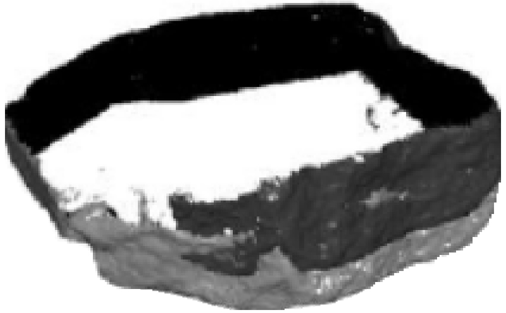
http://jjcao.github.io/DigitalGeometry/

The purpose of computing is insight, not numbers

Last Time

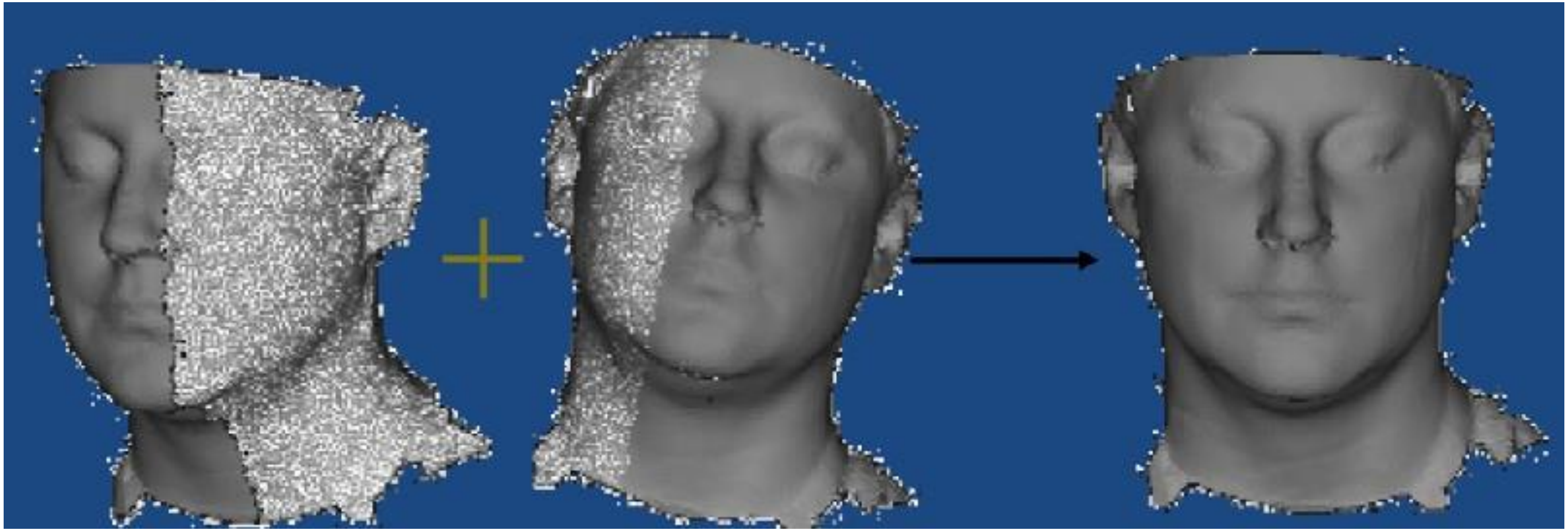
- **Surface Registration**
  - Pairwise ICP & Variants
  - Point-to-point/plane metric
  - BSP closes point search
  - Stability Analysis
  - Global Registration





Shape Matching for Model Alignment

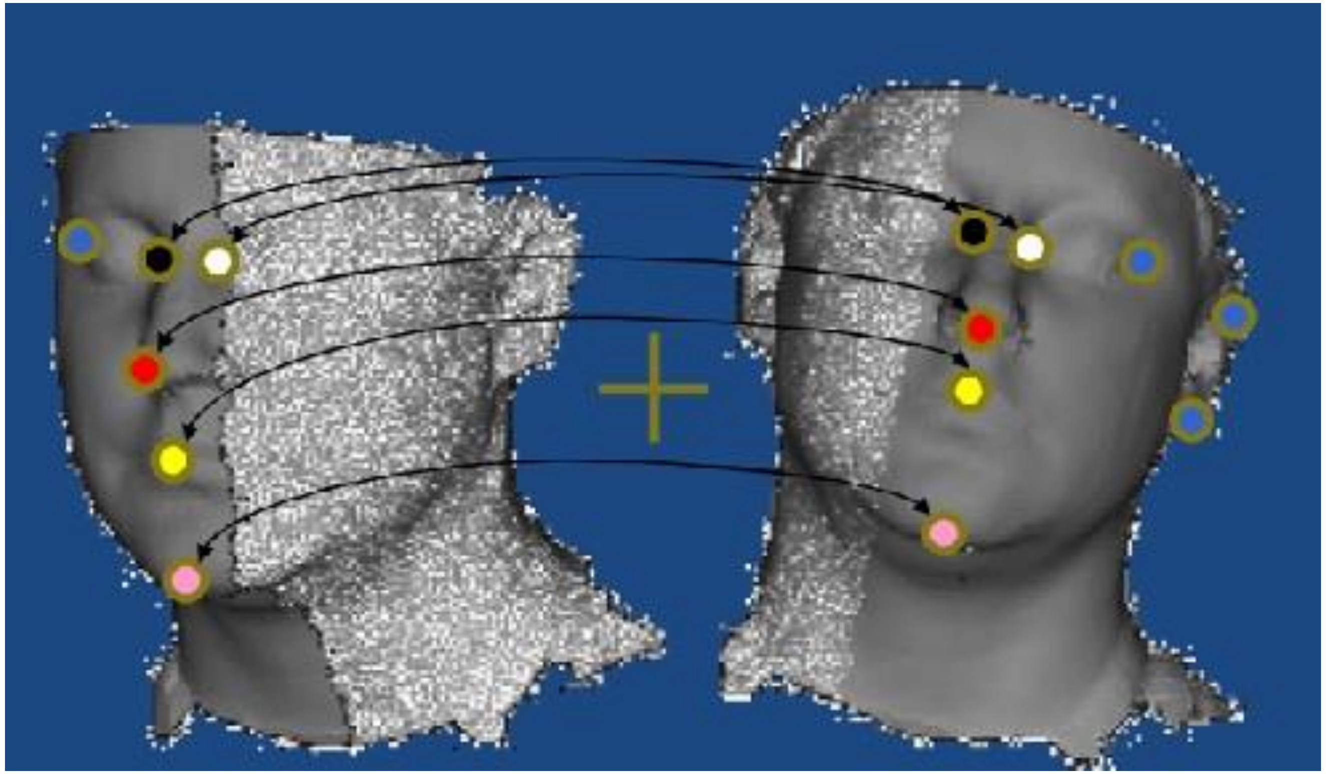
- **Goal**
- Given two partially overlapping scans, compute transformation that aligns the two.
- **No assumption** about rough initial alignment



Partially Overlapping Scans Aligned Scans

Shape Matching for Model Alignment

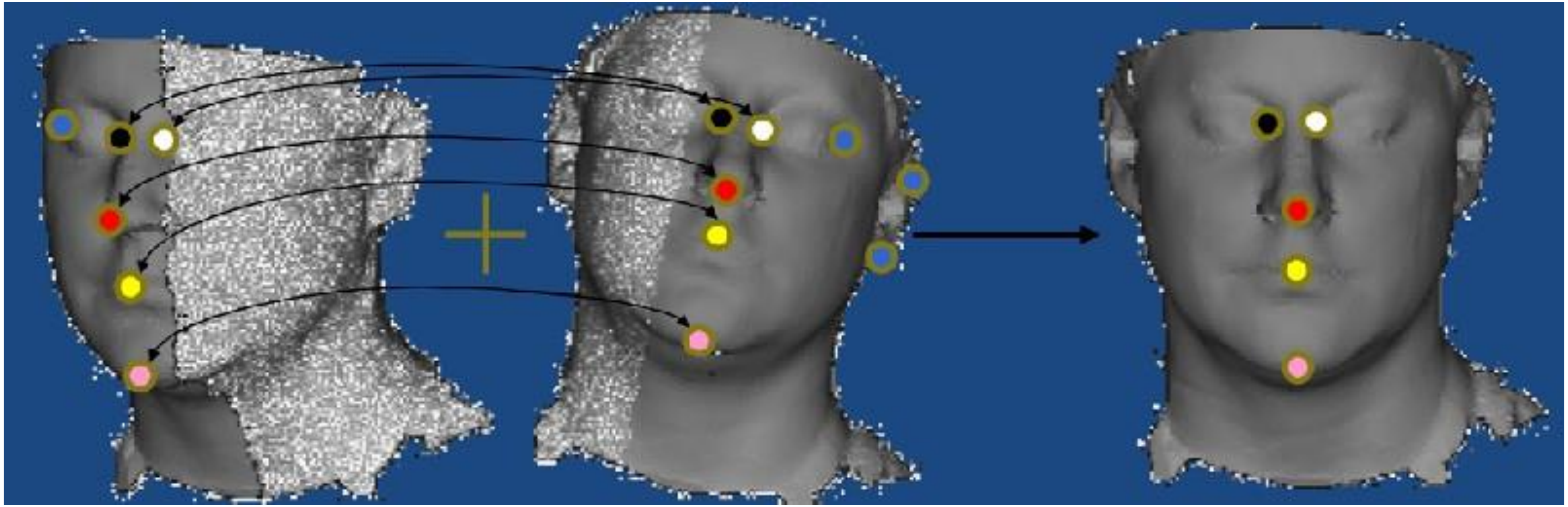
- **Approach**
  - Find feature points on the two scans
  - Establish correspondences



Partially Overlapping Scans

Shape Matching for Model Alignment

- **Approach**
  1. Find feature points on the two scans
  2. Establish correspondences
  3. Compute the alignment



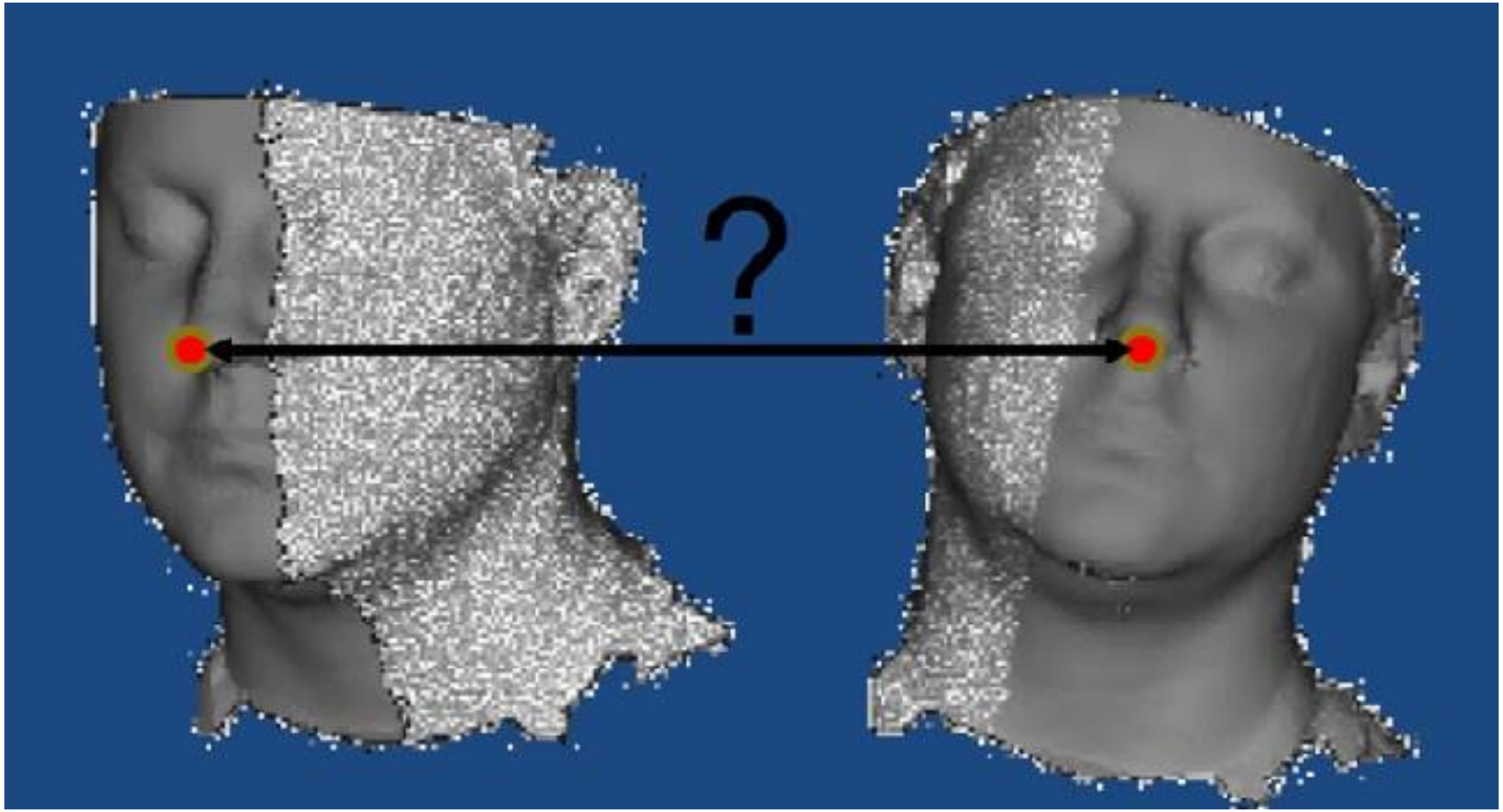
Partially Overlapping Scans **Aligned Scans**

Outline

- Global Shape Correspondence
  - Shape Descriptors
  - Alignment
- Partial Shape Correspondence
  - From Global to Local
  - Pose Normalization
  - Partial Shape Descriptors
- Registration
  - Closed Form Solutions
  - Branch & Bound
  - Random Sample Consensus (RANSAC)

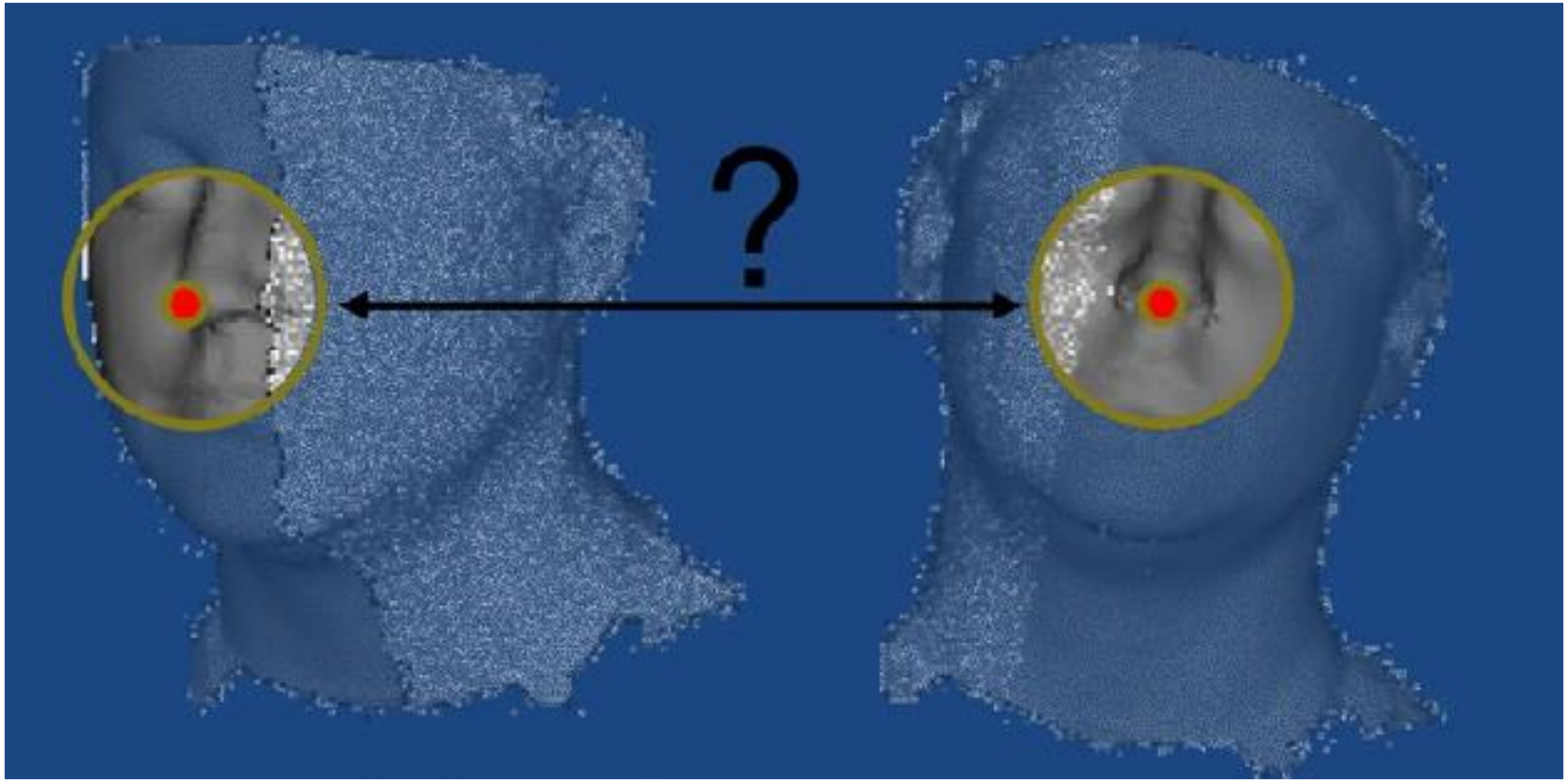
Correspondence

- Goal
  - Identify when two points on different scans represent the same feature



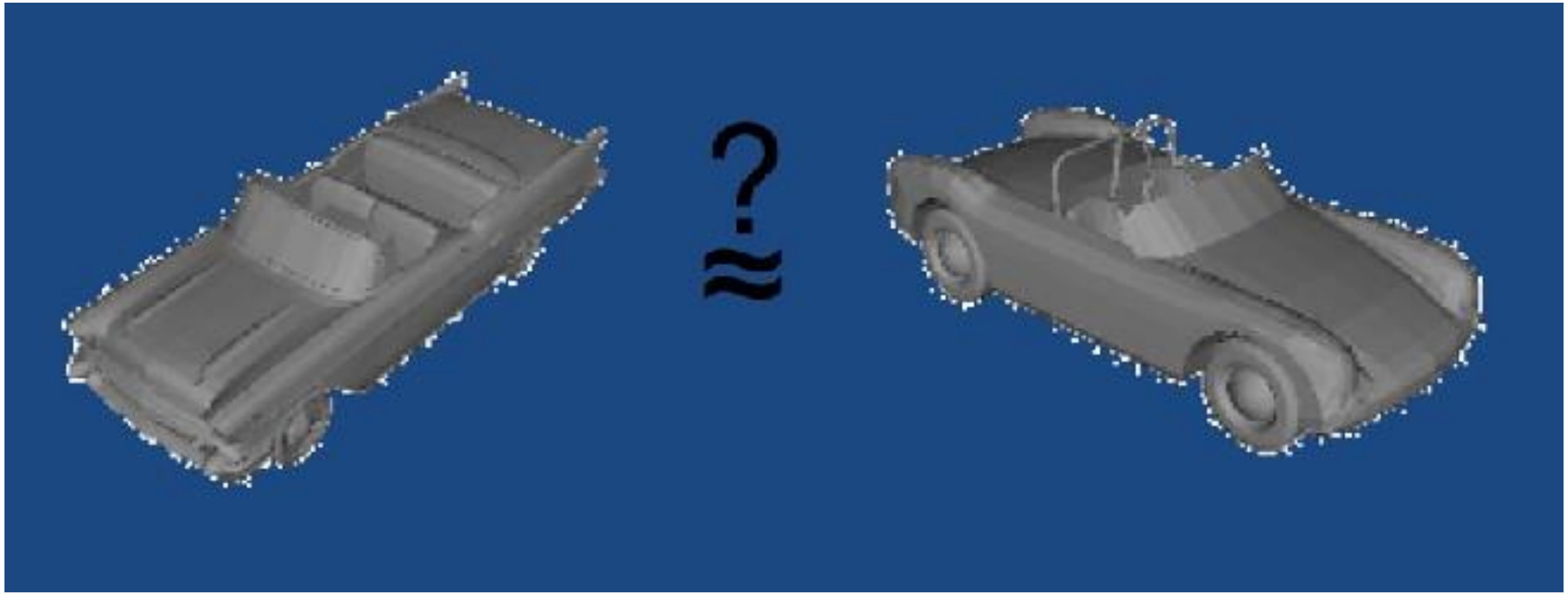
Local **Correspondence**

- Goal
  - Identify when two points on different scans represent the same feature
  - Are the surrounding regions similar?



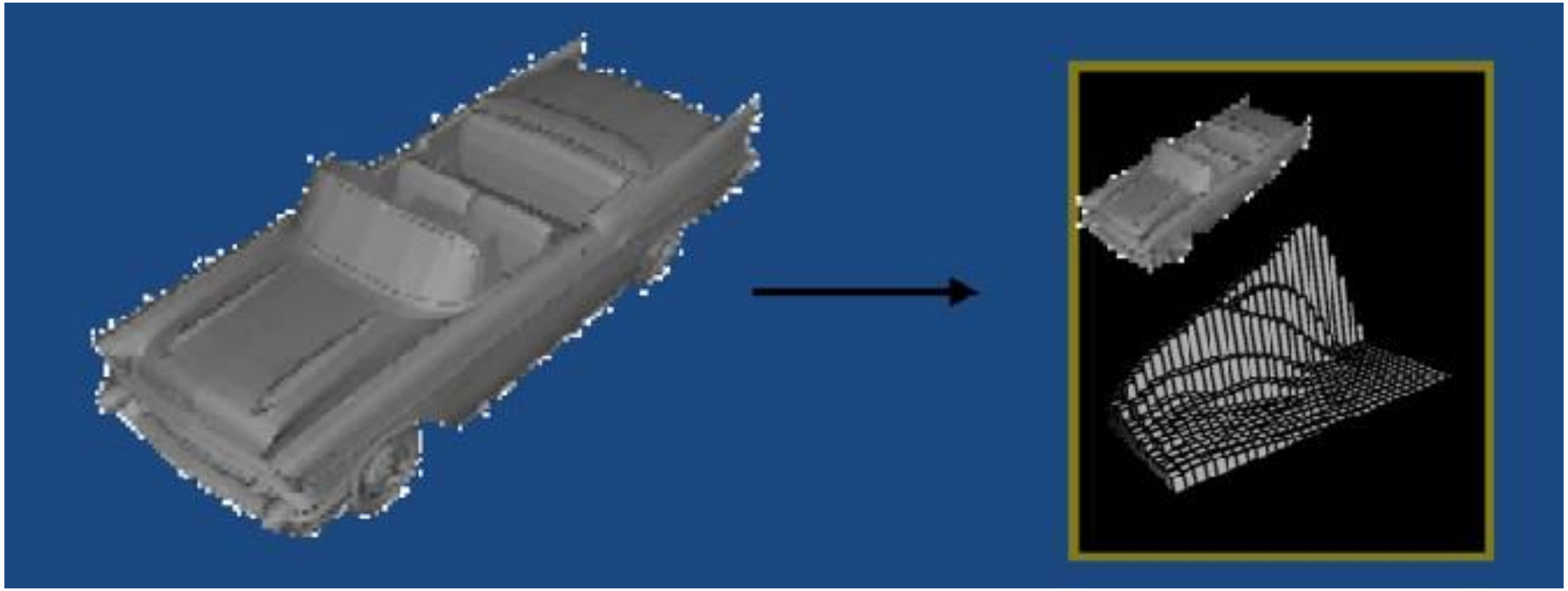
Global **Correspondence**

- More Generally:
  - Given two models, determine if they represent the same/similar shapes
    - models can have different representations, tesselations, topologies, etc.



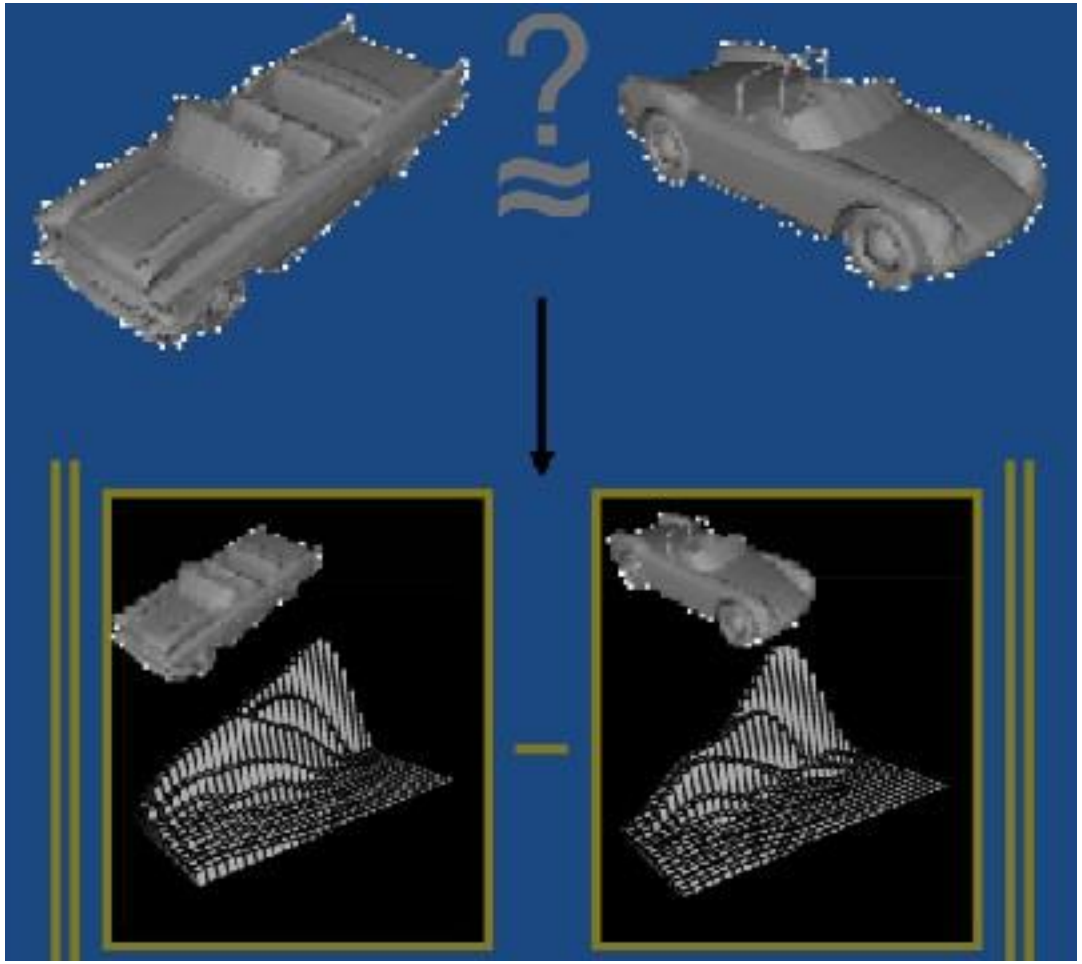
Global **Correspondence**

- **Approach:**
- Represent each model by a shape descriptor:
  - A **structured** abstraction of a 3D model
  - that captures **salient** shape information



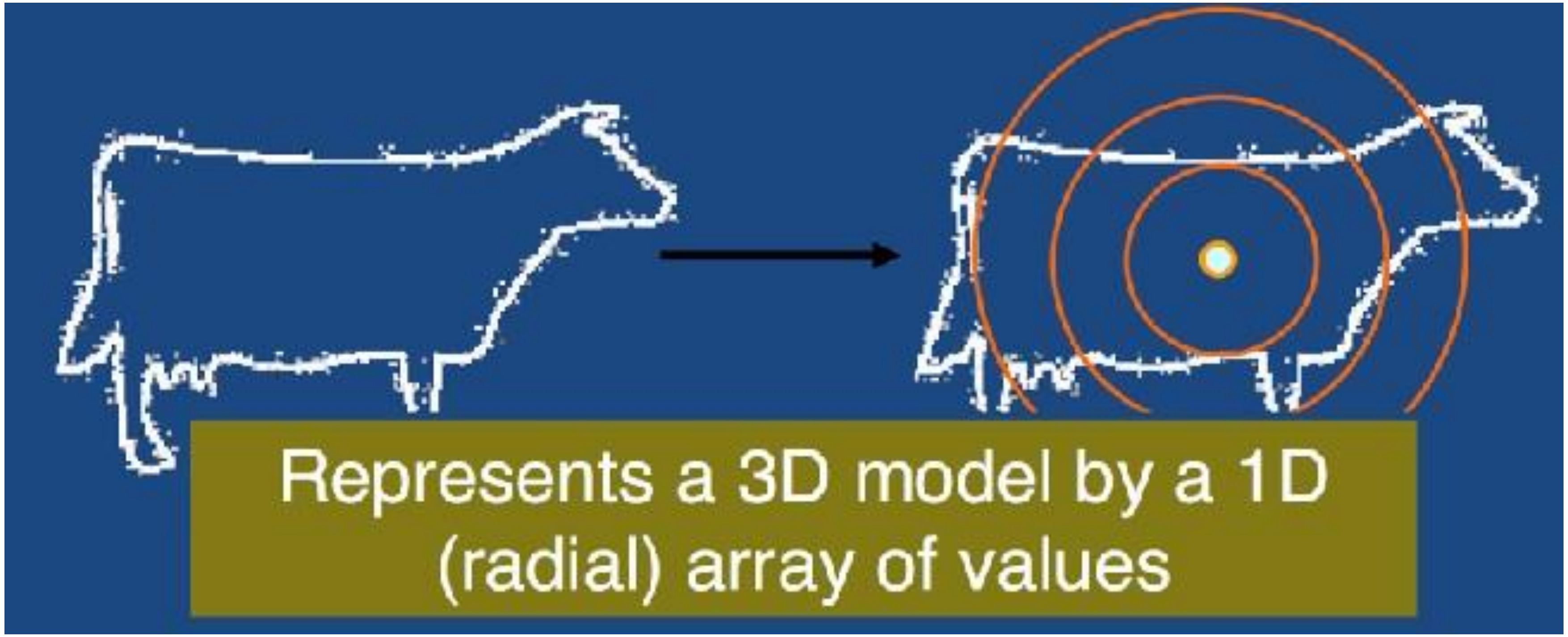
Global **Correspondence**

- **Approach:**
  - Represent each model by a shape descriptor
  - Compare shapes by comparing their shape descriptors



Shape Descriptors: Examples

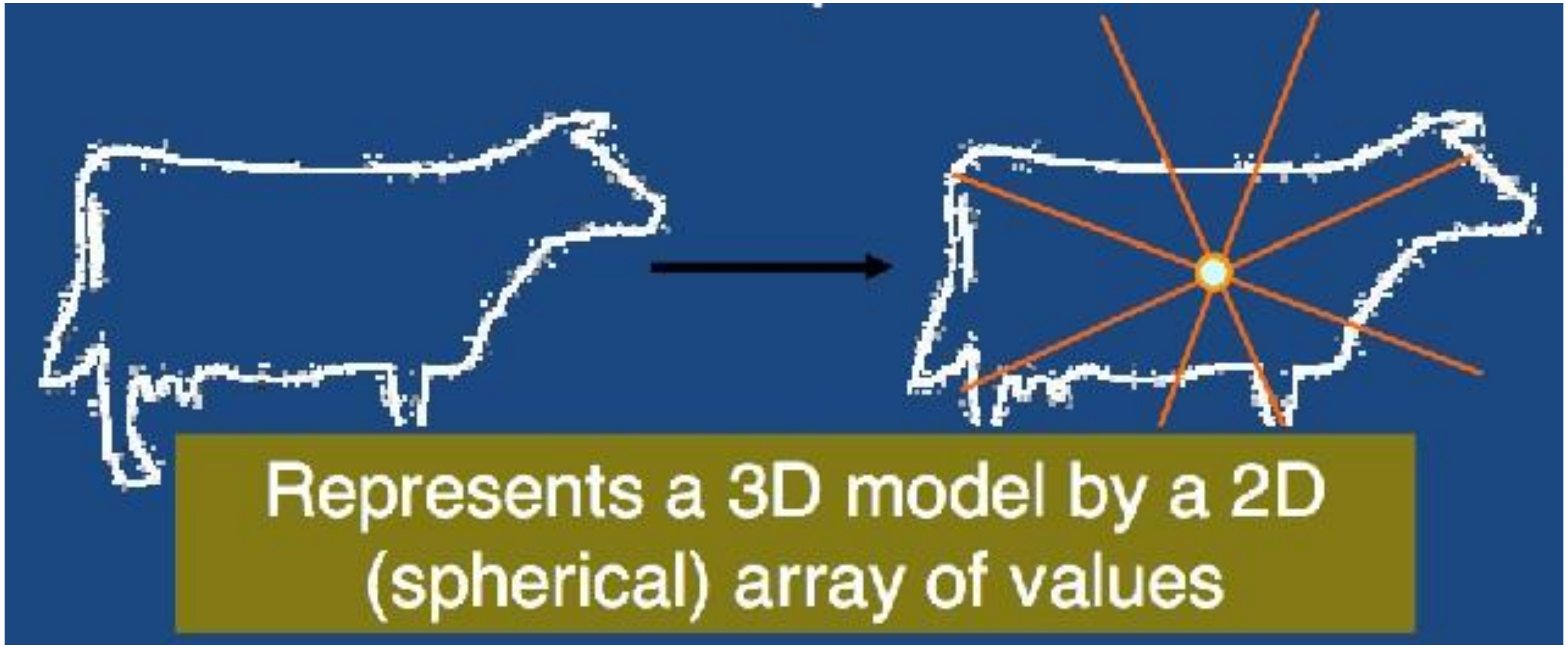
- Shape descriptor stores a histogram of how much surface **area** resides within different **concentric shells** in space



[Ankerst et al. 1999]

Shape Descriptors: Examples

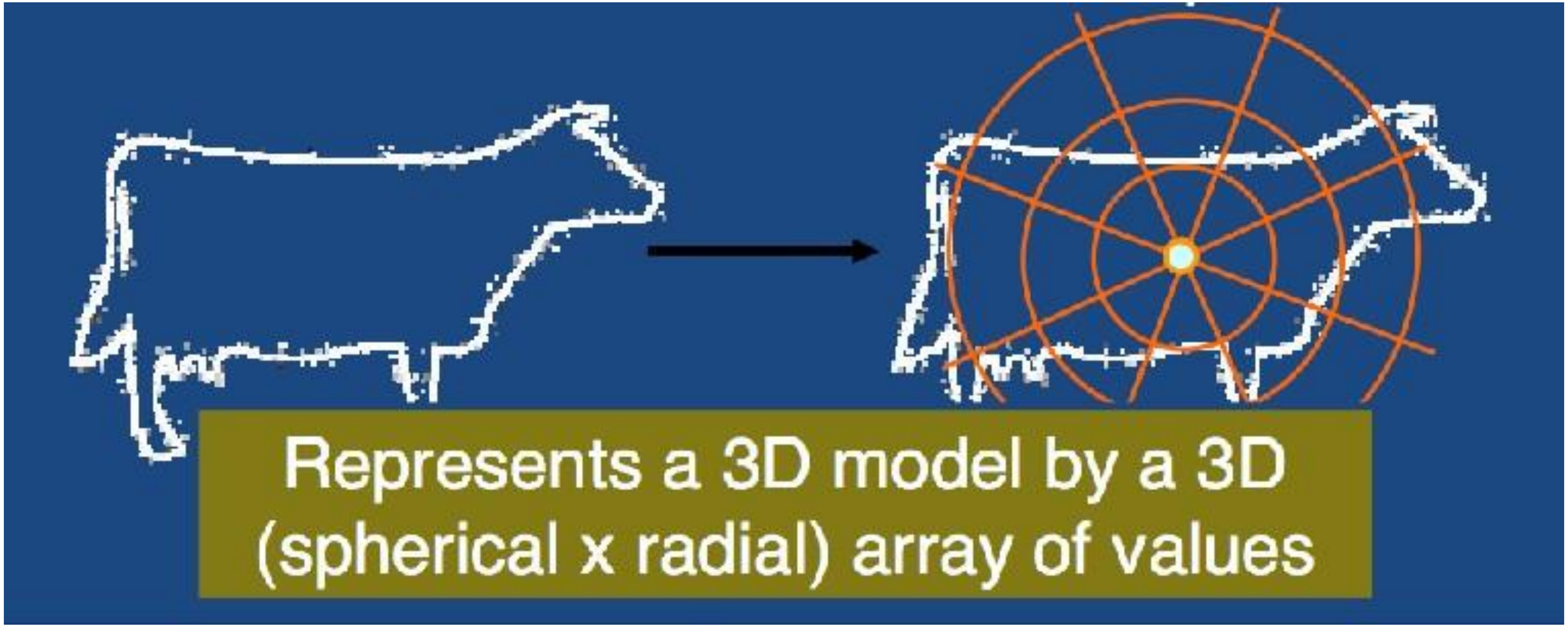
- Shape descriptor stores a histogram of how much surface **area** resides within different **sectors** in space



[Ankerst et al. 1999]

Shape Descriptors: Examples

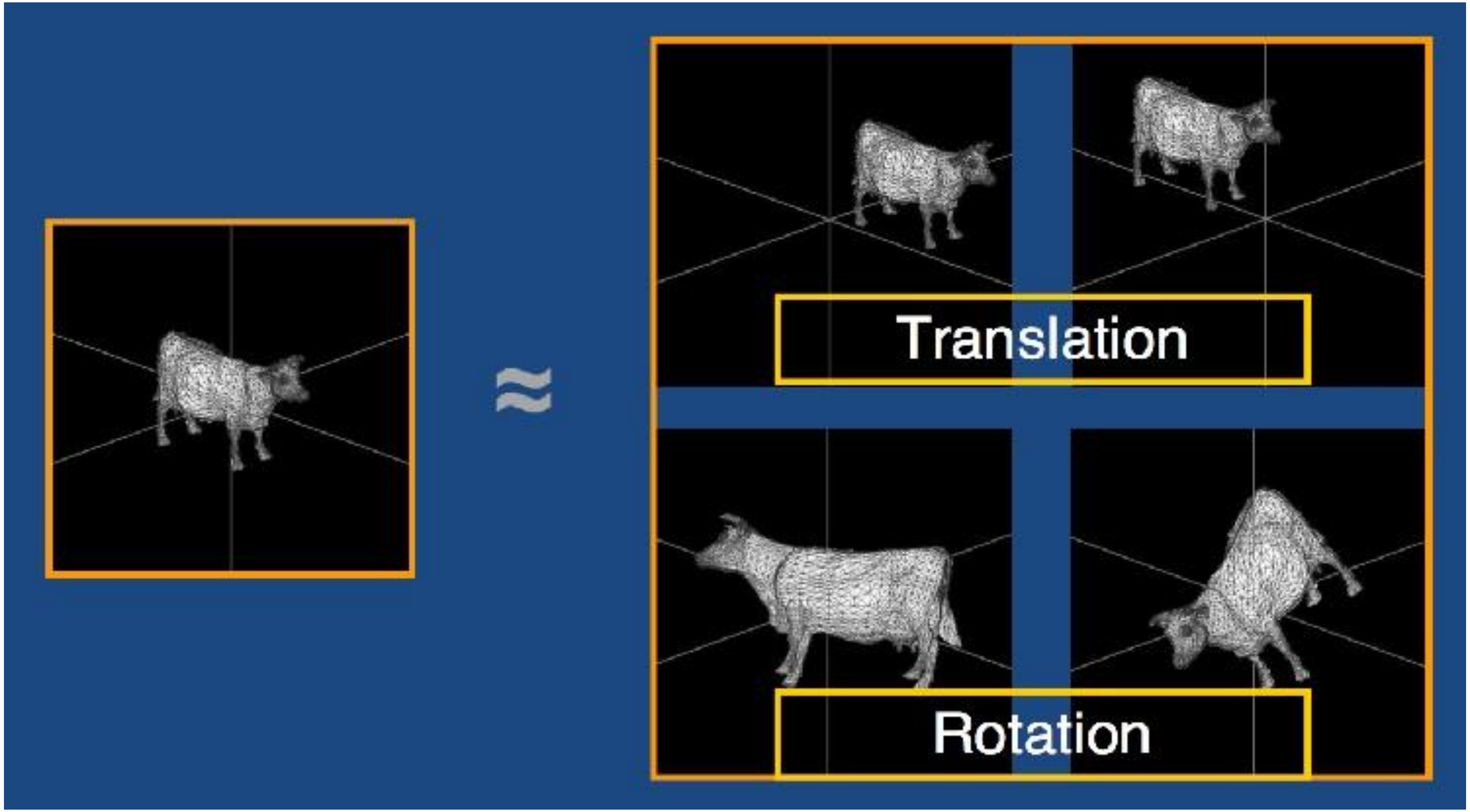
- Shape descriptor stores a histogram of how much surface **area** resides within different **shells** and **sectors** in space



[Ankerst et al. 1999]

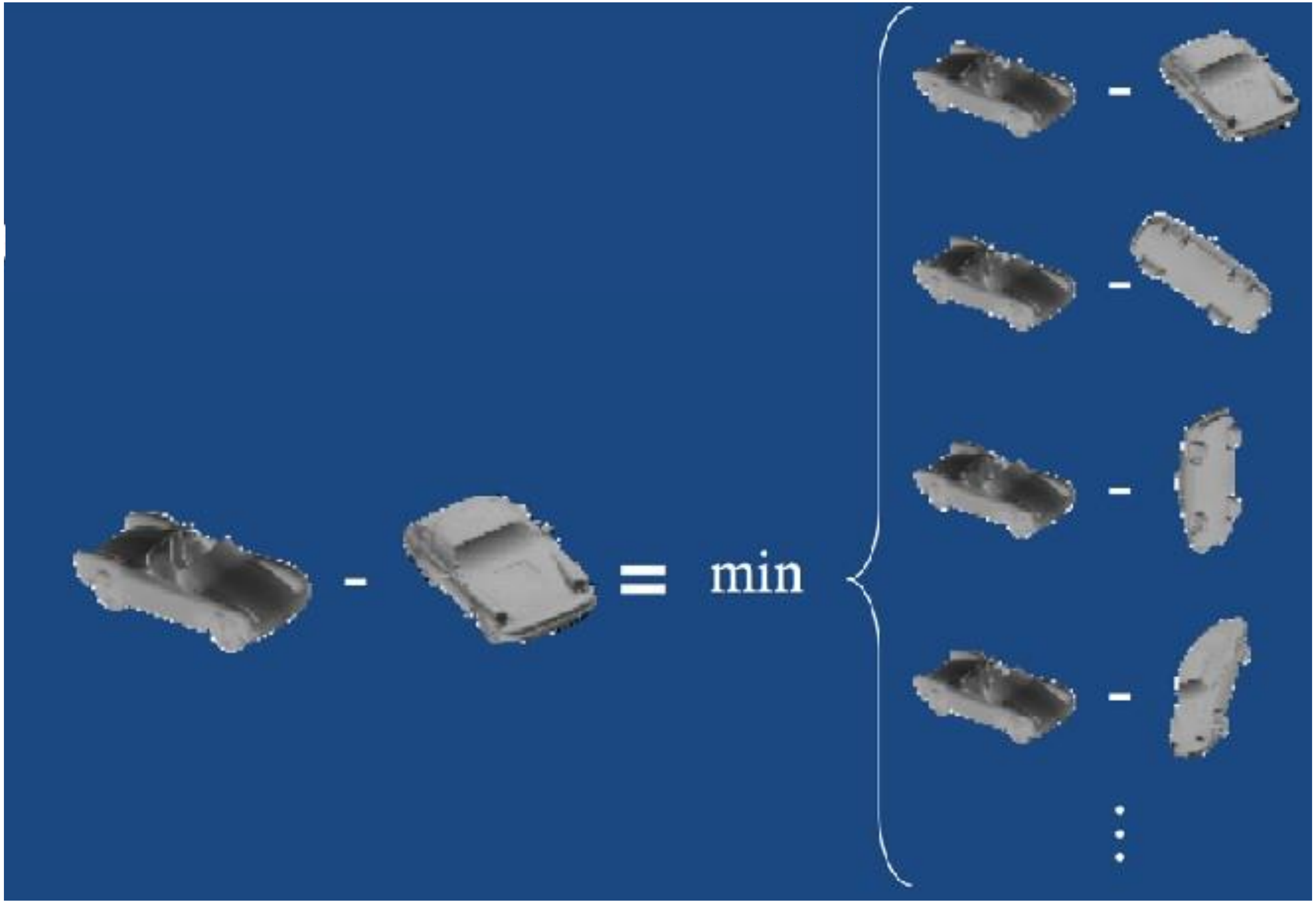
Shape Descriptors: Challenge

- The **shape** of a model does not change when a rigid body transformation is applied to the model



Shape Descriptors: Challenge

- To compare two models, we need them at their optimal alignment

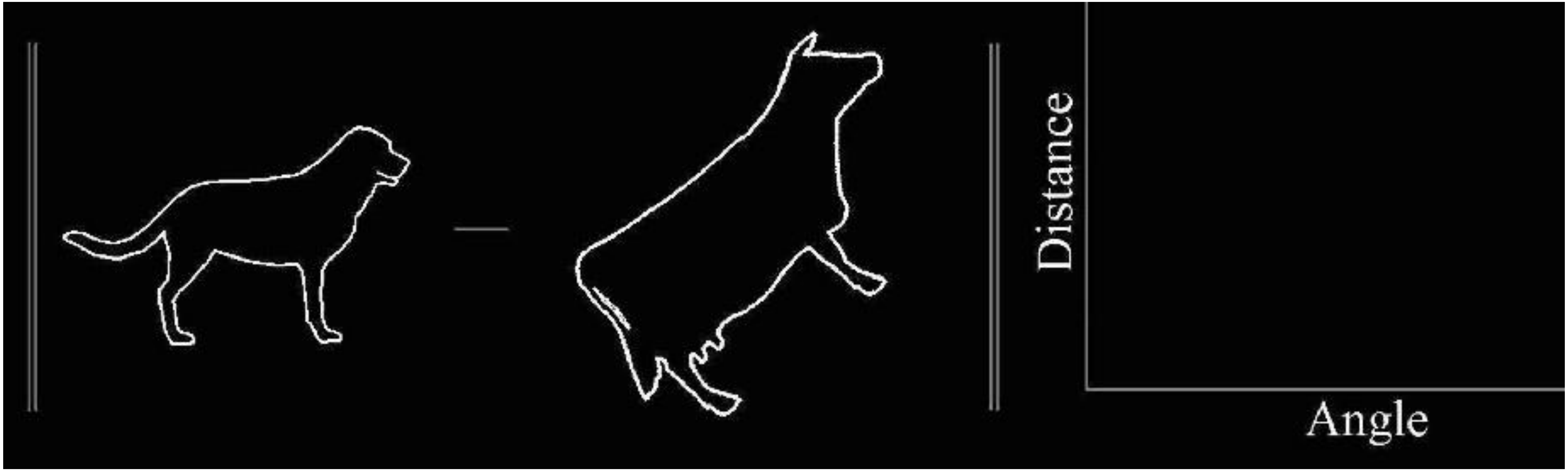


Shape Descriptors: Alignment

- Three general methods:
  - Exhaustive Search
  - Normalization
  - Invariance

Shape Descriptors: Alignment

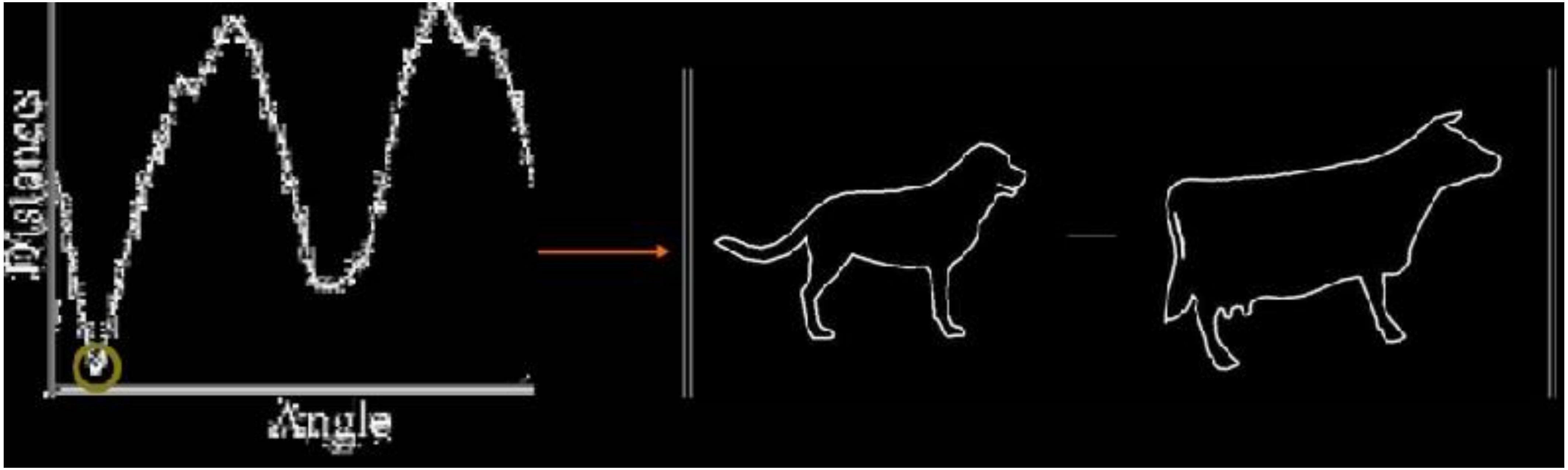
- **Exhaustive Search:**
  - Compare at all alignments



Exhaustive search for optimal rotation

Shape Descriptors: Alignment

- **Exhaustive Search:**
  - Compare at all alignments
  - Correspondence is determined by the alignment at which the models are closest



Exhaustive search for optimal rotation

Shape Descriptors: Alignment

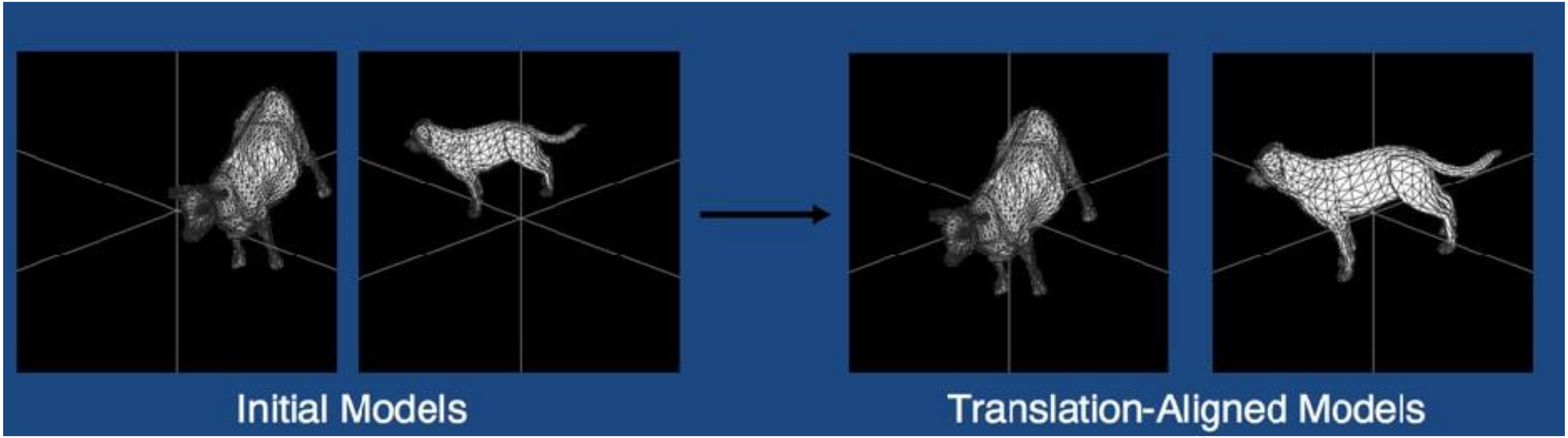
- **Exhaustive Search:**
  - Compare at all alignments
  - Correspondence is determined by the alignment at which the models are closest

- **Properties:**
  - Gives the correct answer (w.r.t. the metric)
  - While slow on a single processor, it can be parallelized (Clusters? Multi-Threading? GPU?)

Shape Descriptors: Alignment

Normalization:

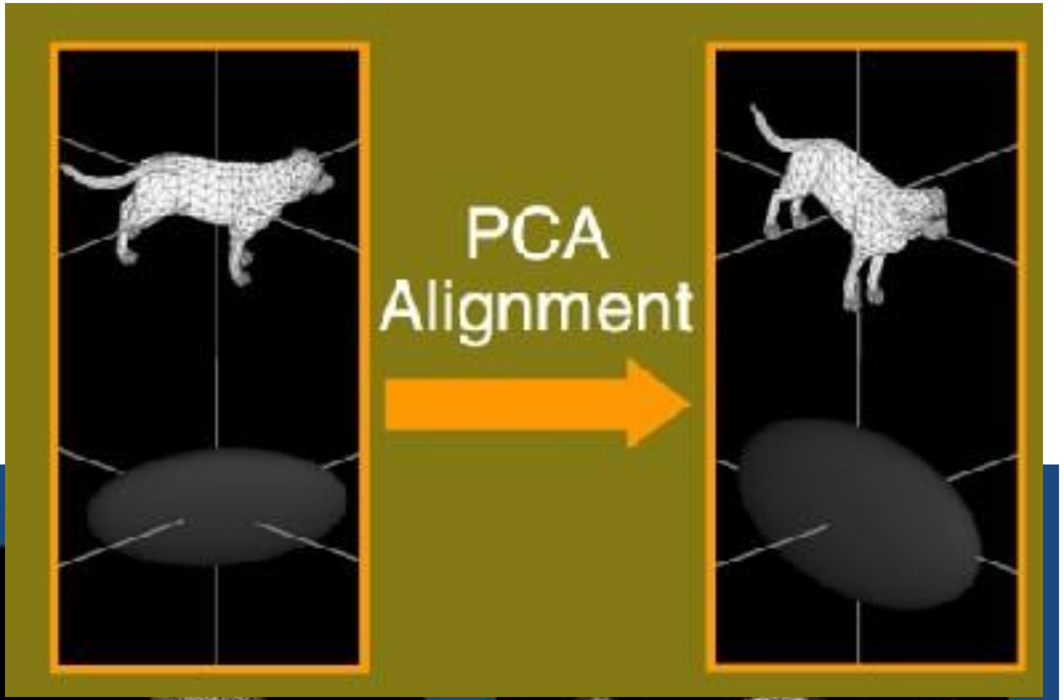
- Put each model into a canonical frame:
  - Translation：Center of Mass
  - Rotation

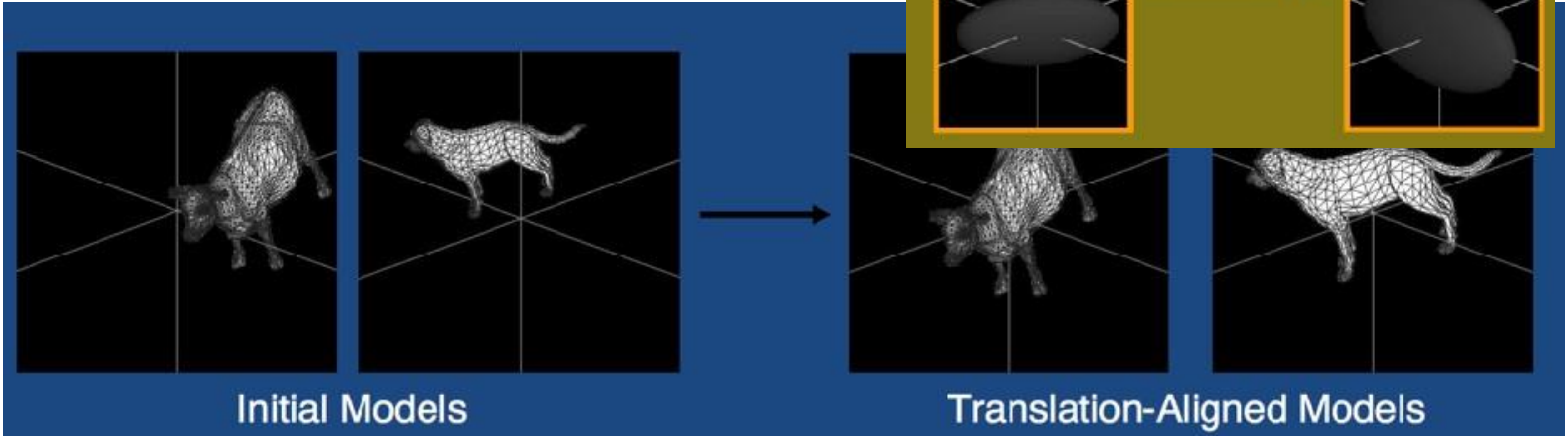


Shape Descriptors: Alignment

Normalization:

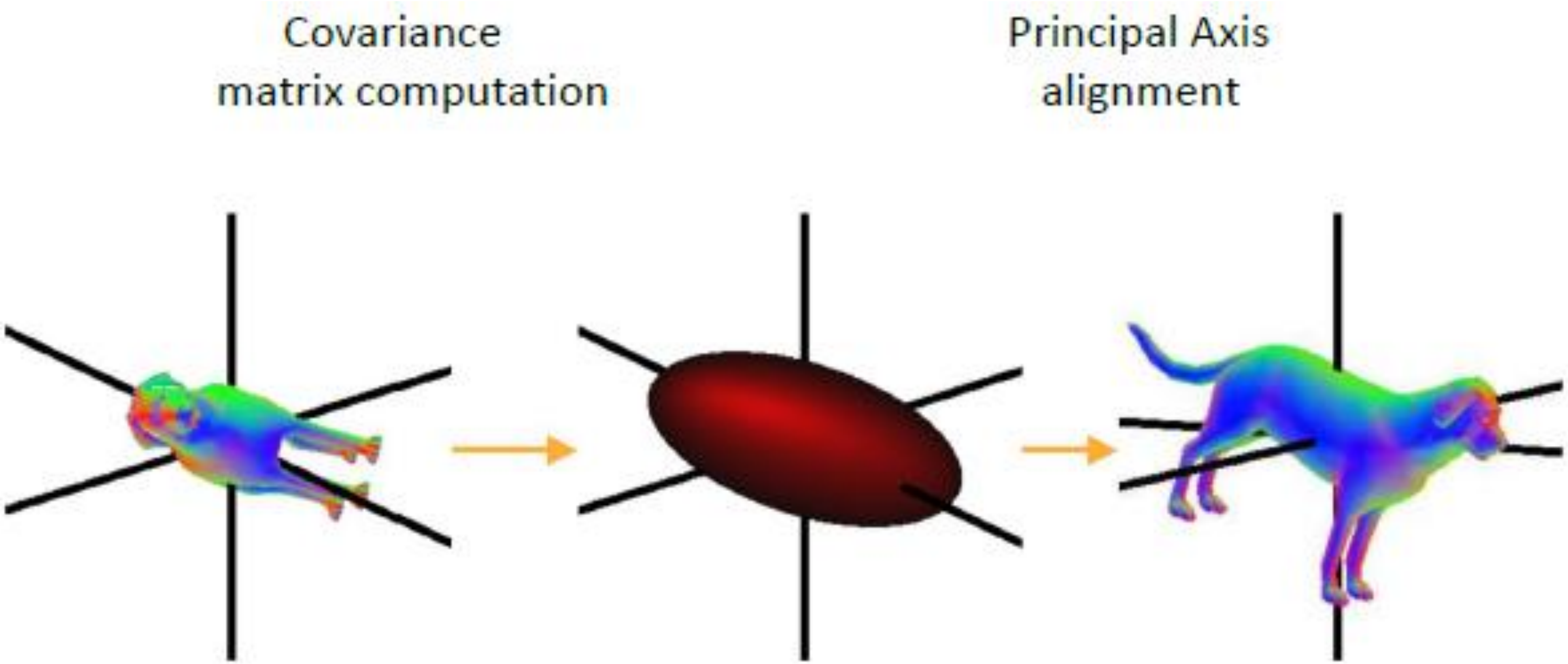
- Put each model into a canonical frame:
  - Translation： Center of Mass
  - Rotation: PCA alignment





Coarse alignment – PCA

- Use PCA to place models into a canonical coordinate frame



Principal axis computation

- Given a collection of points {pi}, form the co-variance matrix:

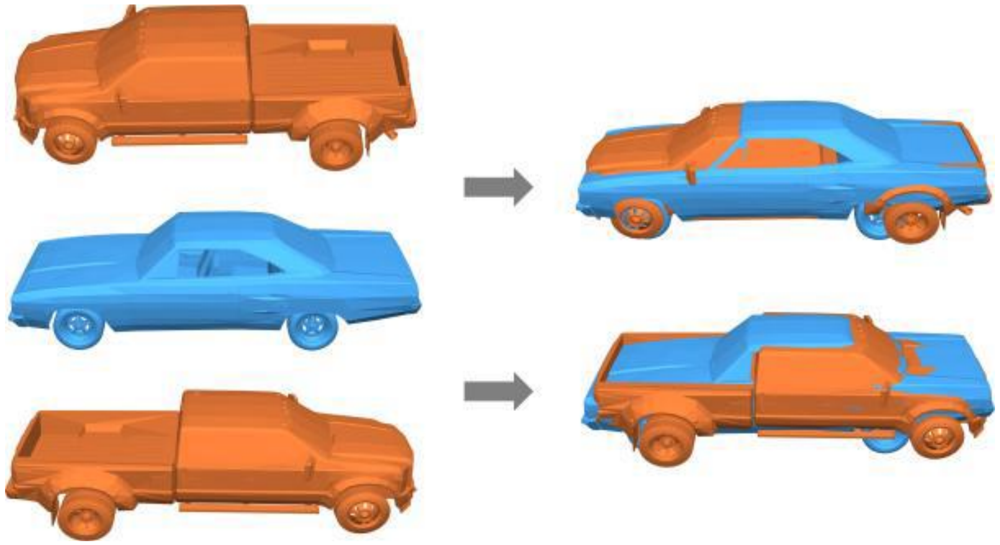
$$\mathbf{c} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{p}_i$$

$$C = \frac{1}{N}\sum_{i=1}^{N} \mathbf{p}_i\mathbf{p}_i^T - \mathbf{c}\mathbf{c}^T$$

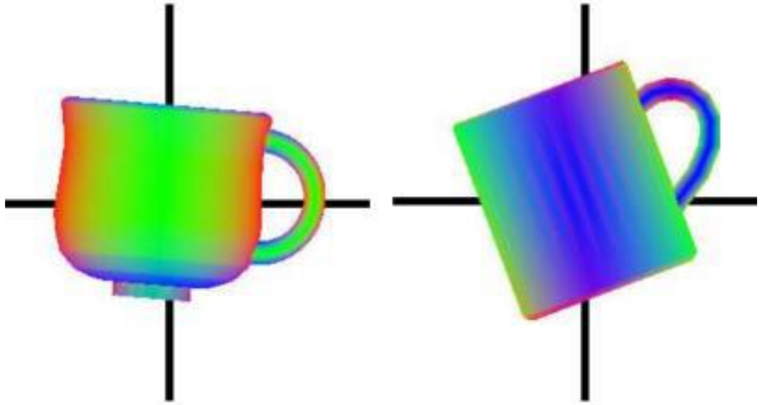
- Compute eigenvectors of matrix C

Issues with PCA

- Principal axes are not oriented



- Axes are unstable when principal values are similar



Shape Descriptors: Alignment

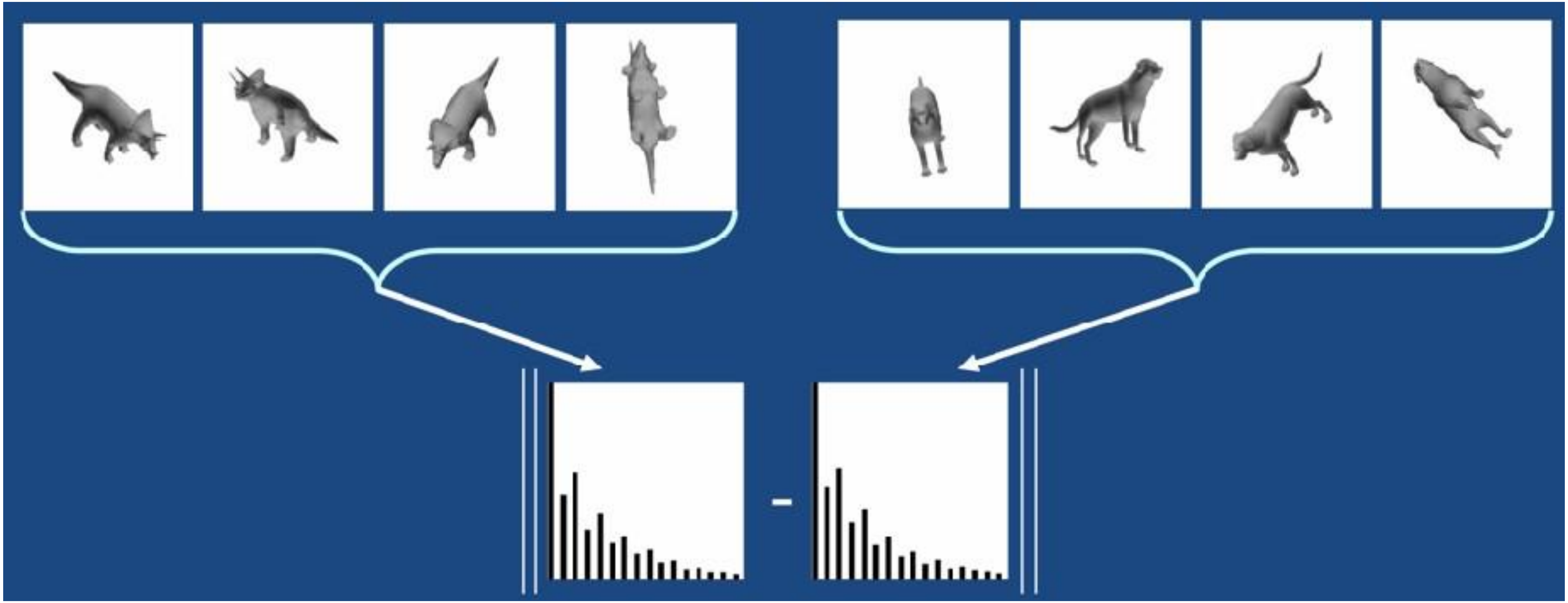
Normalization:

- Put each model into a canonical frame:
    - Translation：Center of Mass
    - Rotation: PCA alignment

- Properties:
    - Efficient
    - Not always robust
    - Not suitable for local feature matching

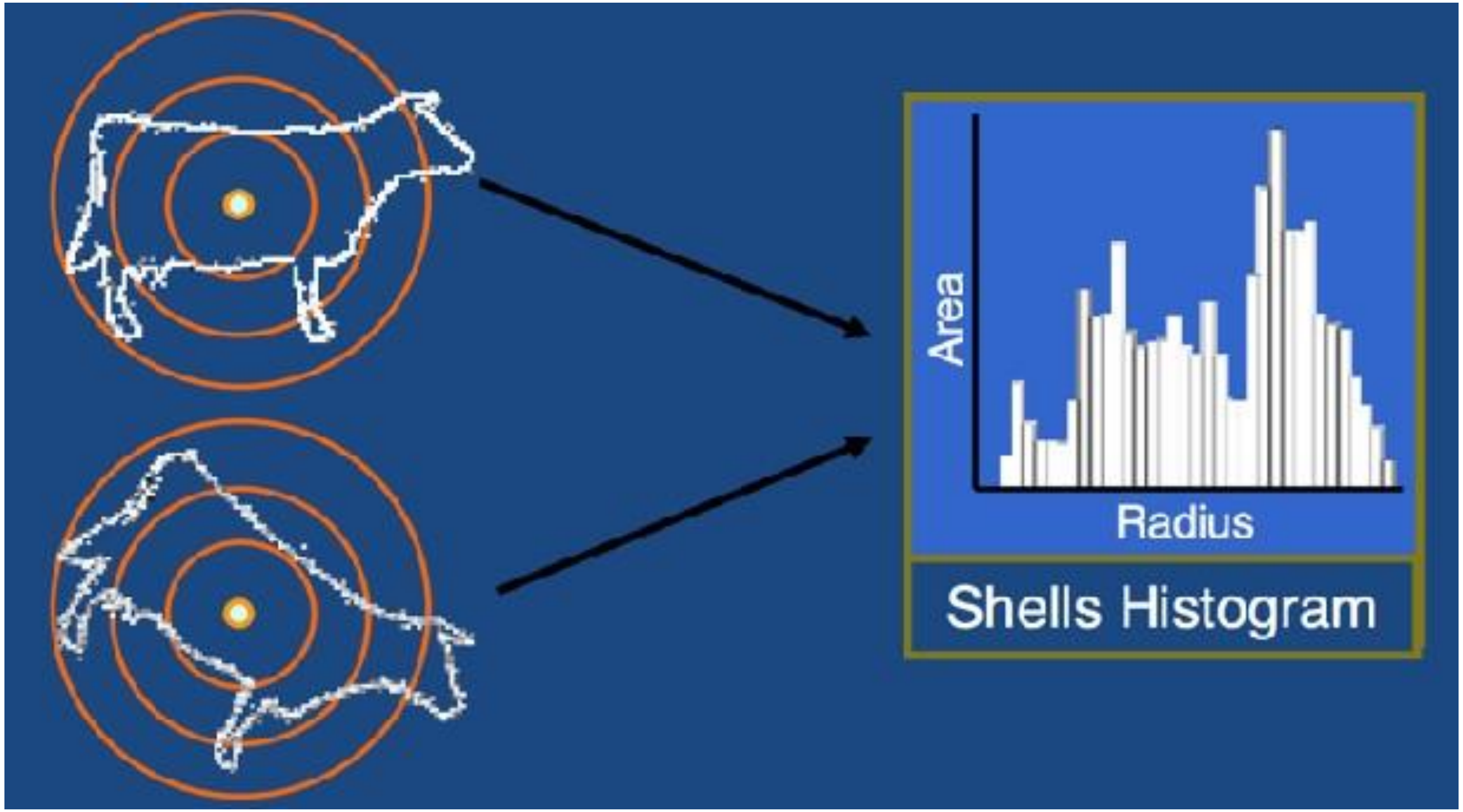
Shape Descriptors: Alignment

- **Invariance:**
  - Represent a model by a shape descriptor that is independent of the pose.



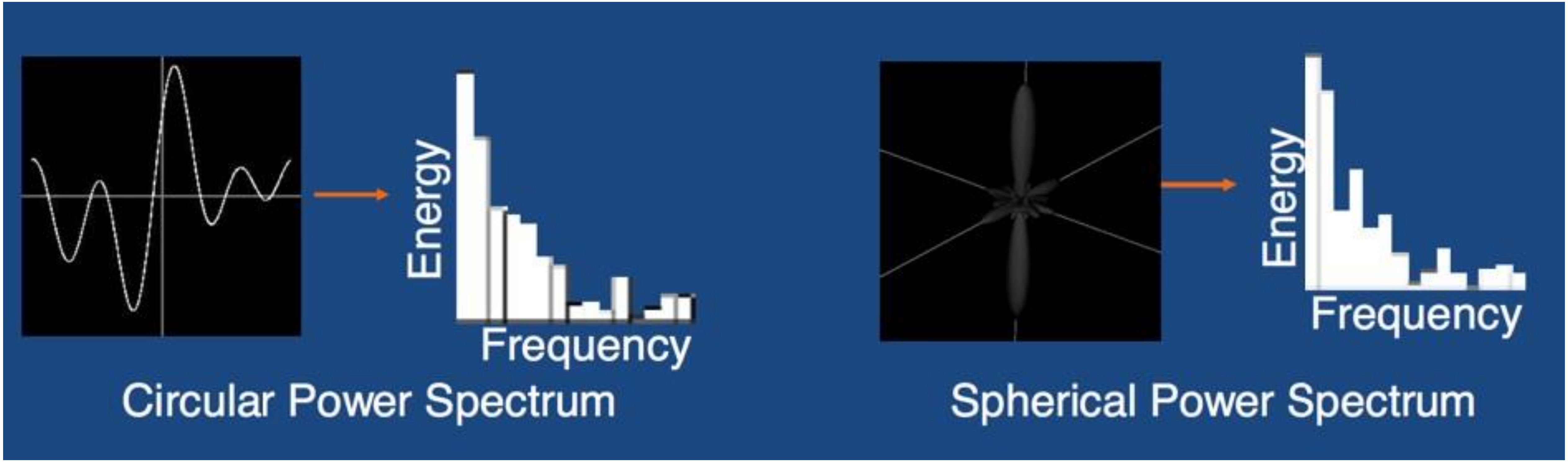
Shape Descriptors: Alignment

- Example: Ankerst’s Shells
  - A histogram of the radial distribution of surface area



Shape Descriptors: Alignment

- Invariance
  - Power spectrum representation
    - Fourier transform for translations
    - Spherical harmonic transform for rotations



storing only the amplitudes of the different frequency components, discarding phase.

Shape Descriptors: Alignment

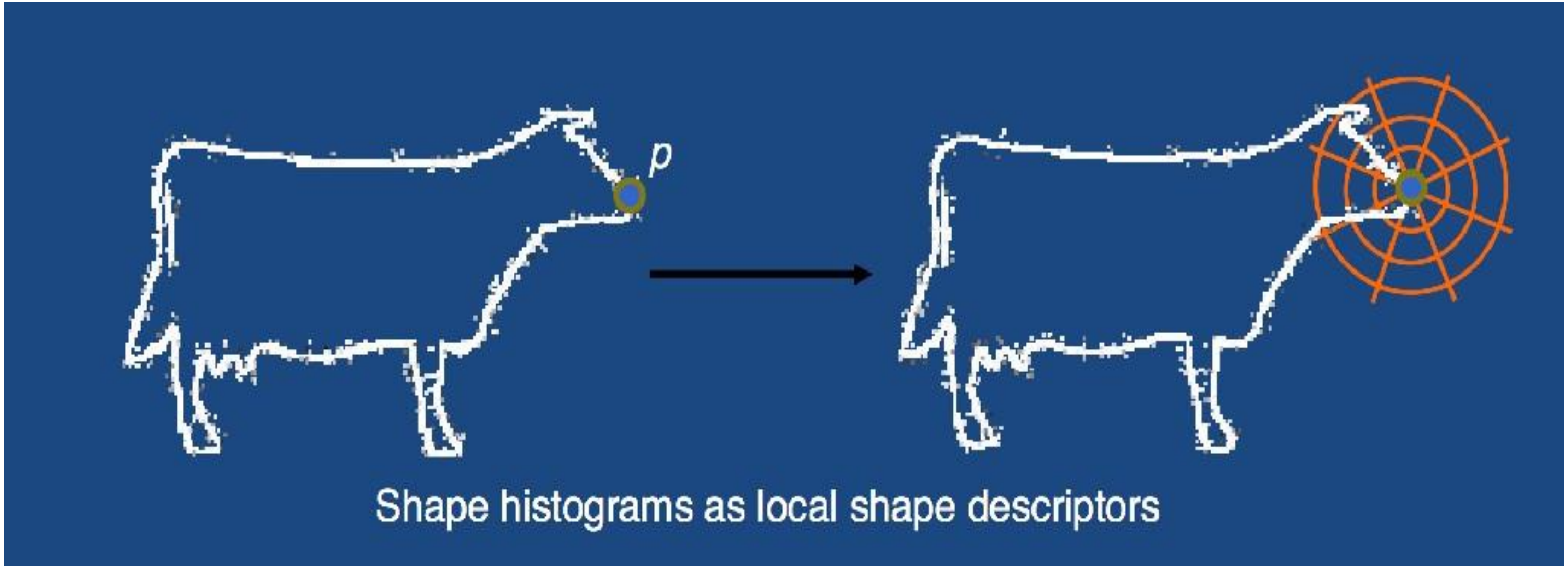
- **Invariance:**
  - Represent a model by a shape descriptor that is independent of the pose
- **Properties:**
  - Compact representation
  - Not always discriminating

Outline

- Global Shape Correspondence
  - Shape Descriptors
  - Alignment
- Partial Shape Correspondence
  - From Global to Local
  - Pose Normalization
  - Partial Shape Descriptors
- Registration
  - Closed Form Solutions
  - Branch & Bound
  - Random Sample Consensus (RANSAC)

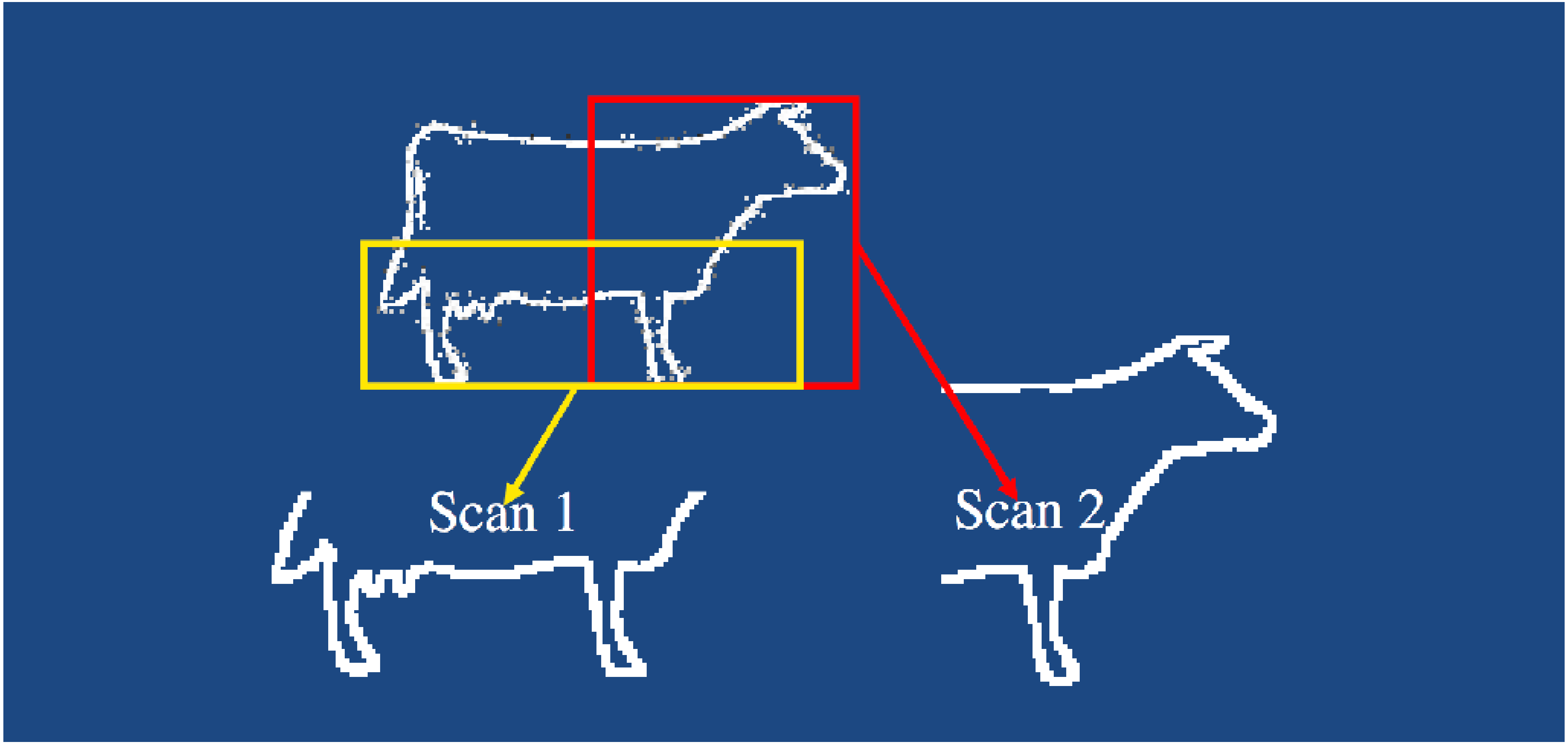
From Global to Local

- To characterize the surface about a point p, take global descriptor and:
  - center it about p (instead of center of mass), and
  - restrict the extent to a small region about p



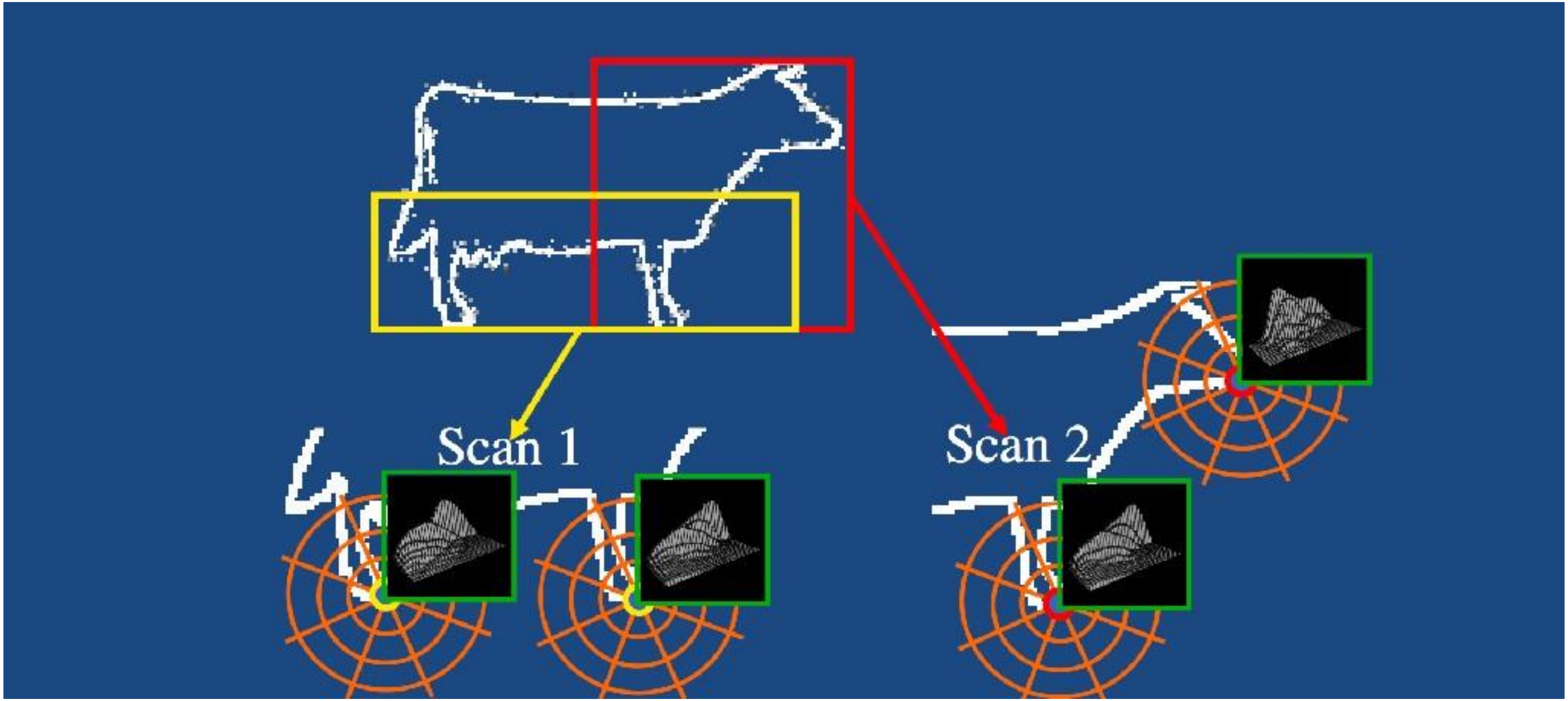
From Global to Local

- **Given scans of a model:**



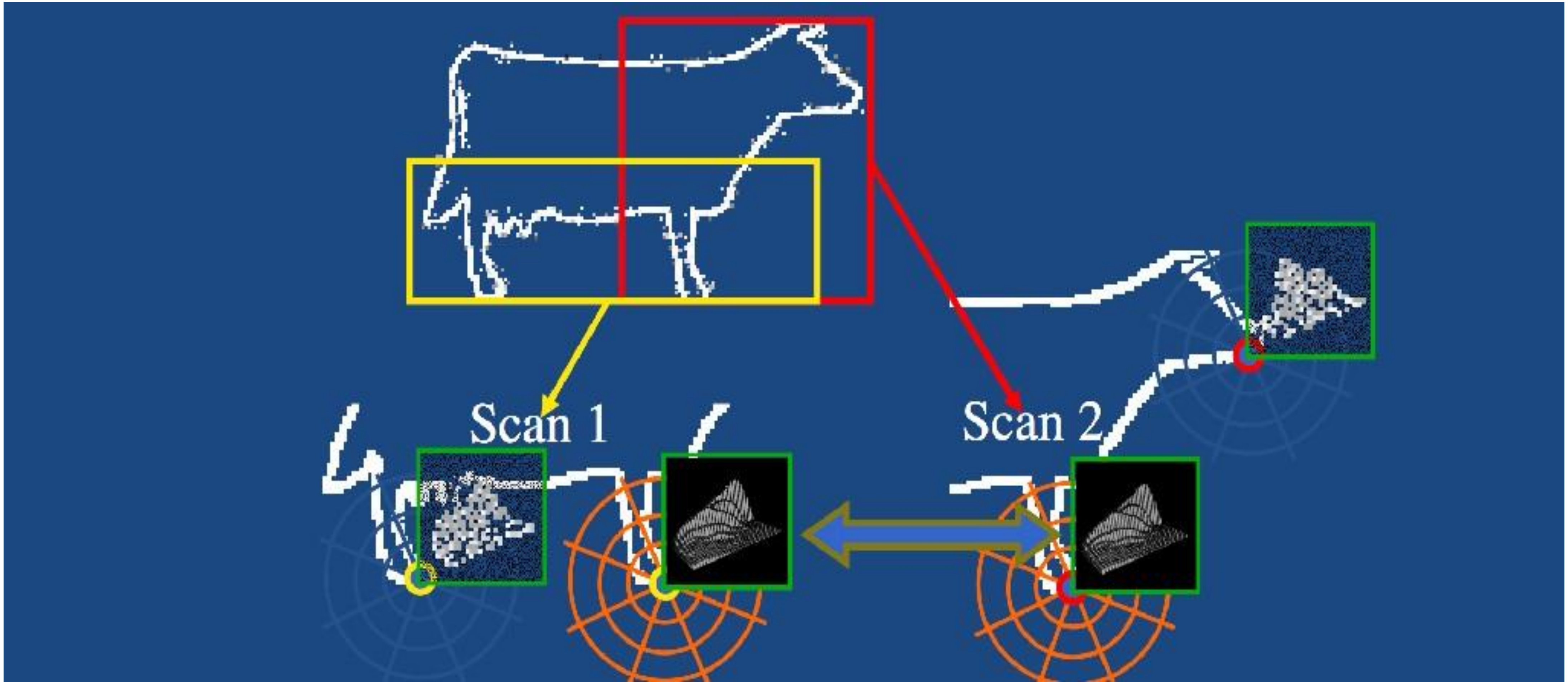
From Global to Local

- Identify the features
- Computer a local descriptor for each feature



From Global to Local

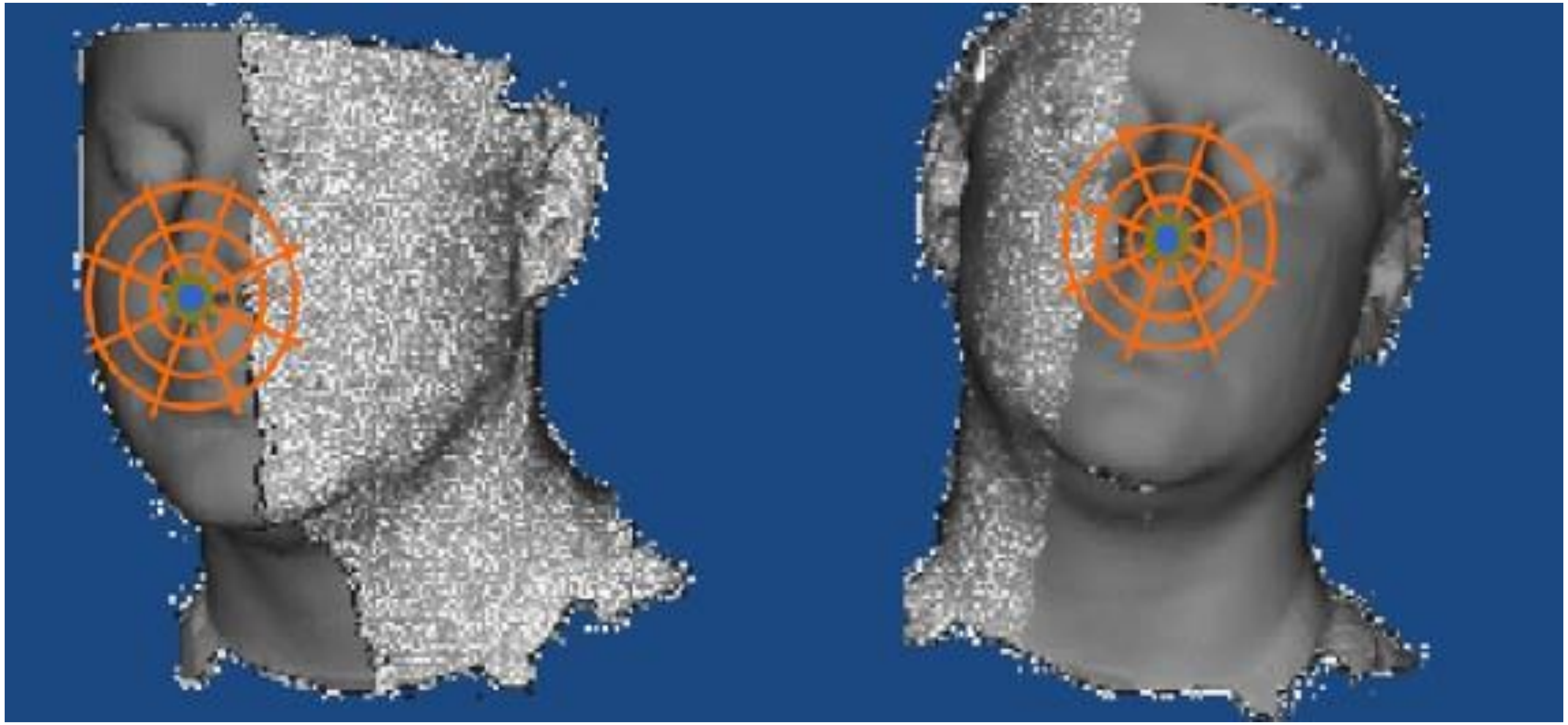
- Identify the features
- Computer a local descriptor for each feature
- Feature correspond → descriptors are similar



Pose Normalization

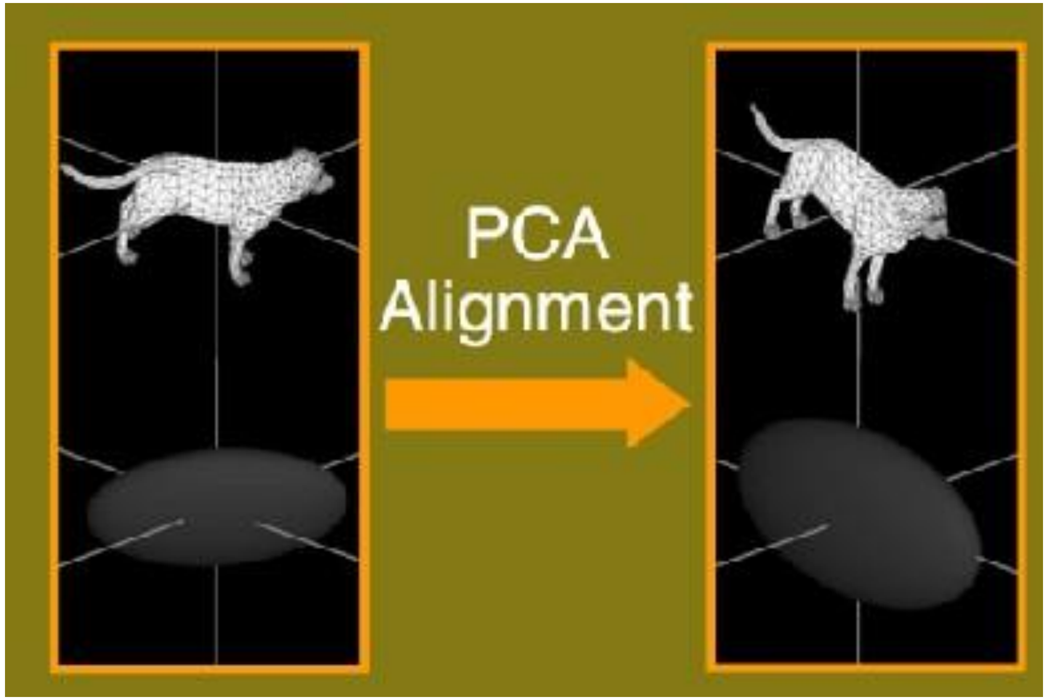
Optimal Transportation is better

- **From Global to Local**
  - Translation: Accounted for by centering the descriptor at the point of interest.
  - Rotation: We still need to be able to match descriptors across different rotations.



Pose Normalization

- **Challenge**
  - Since only parts of the models are given, we cannot use global normalization to align the local descriptors

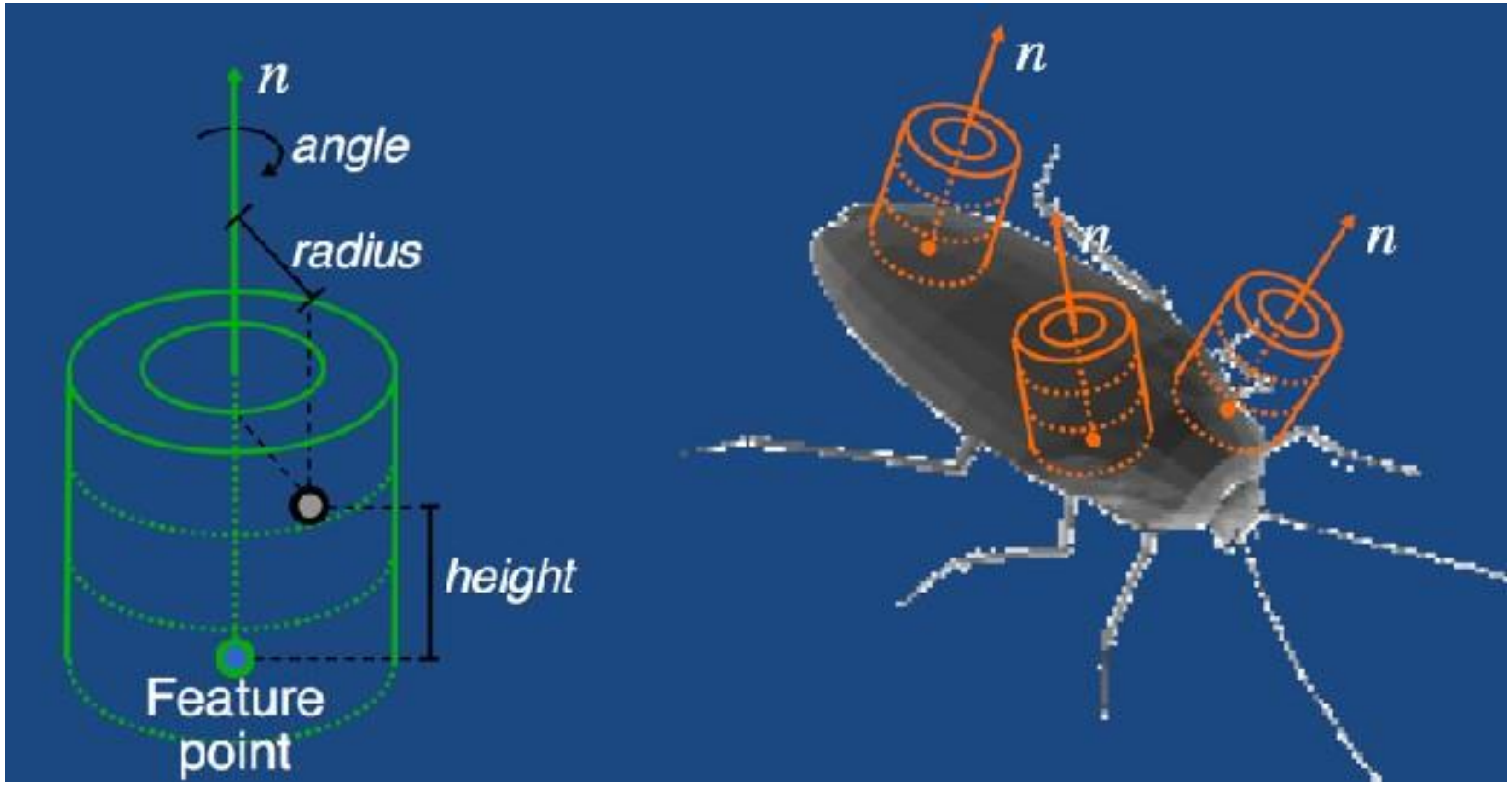


- **Solutions**

Normalize using local information?

Local Descriptors: Examples

- **Variations of Shape Histograms**
  - For each feature, represent its local geometry in cylindrical coordinates about the normal

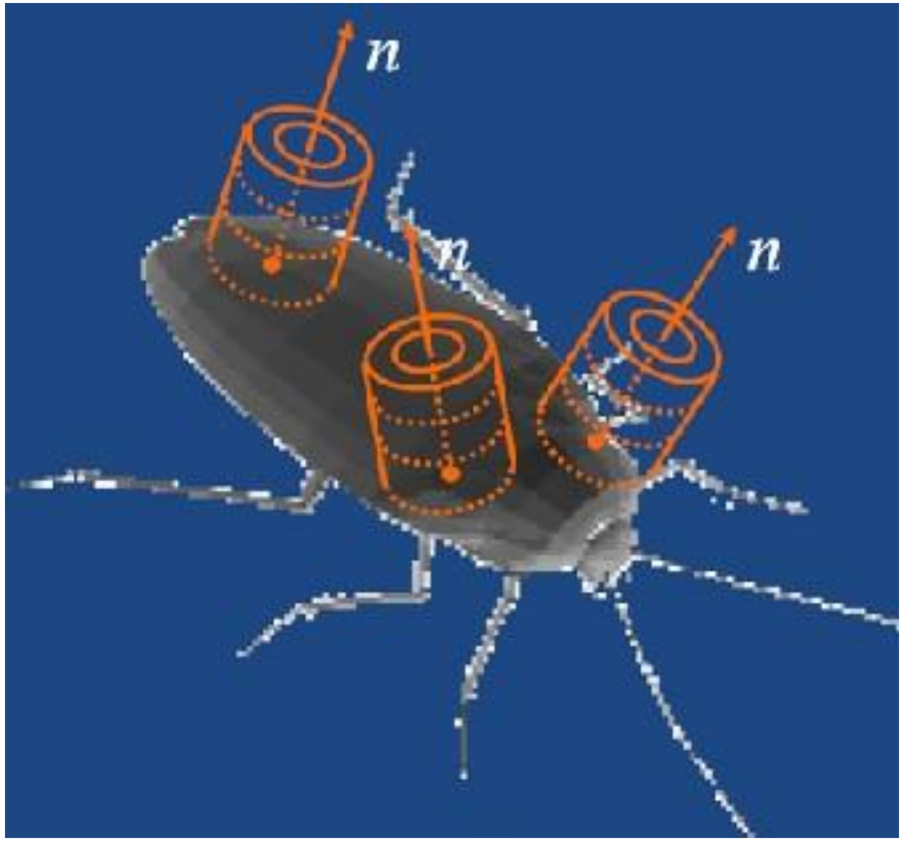


Since the surface normal is consistent across corresponding feature points, the height and radius are in normalized coordinates. However, there is no normalization for the angle about the normal,

Local Descriptors: Examples

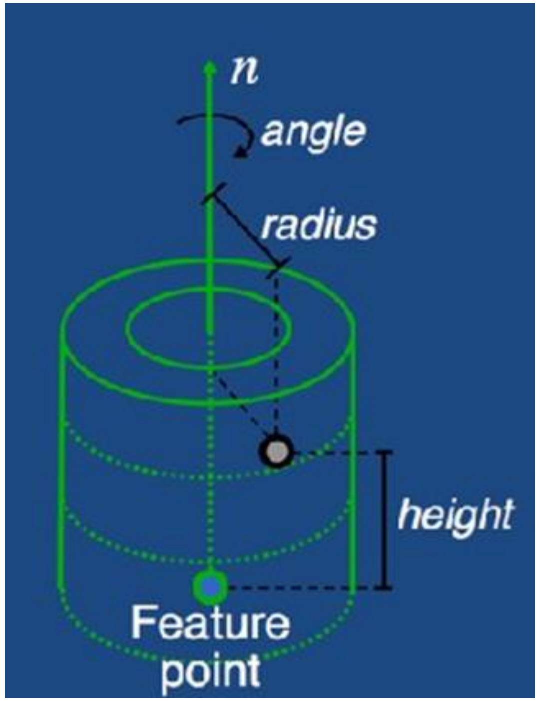
- **Variations of Shape Histograms**
  - For each feature, represent its local geometry in cylindrical coordinates about the normal

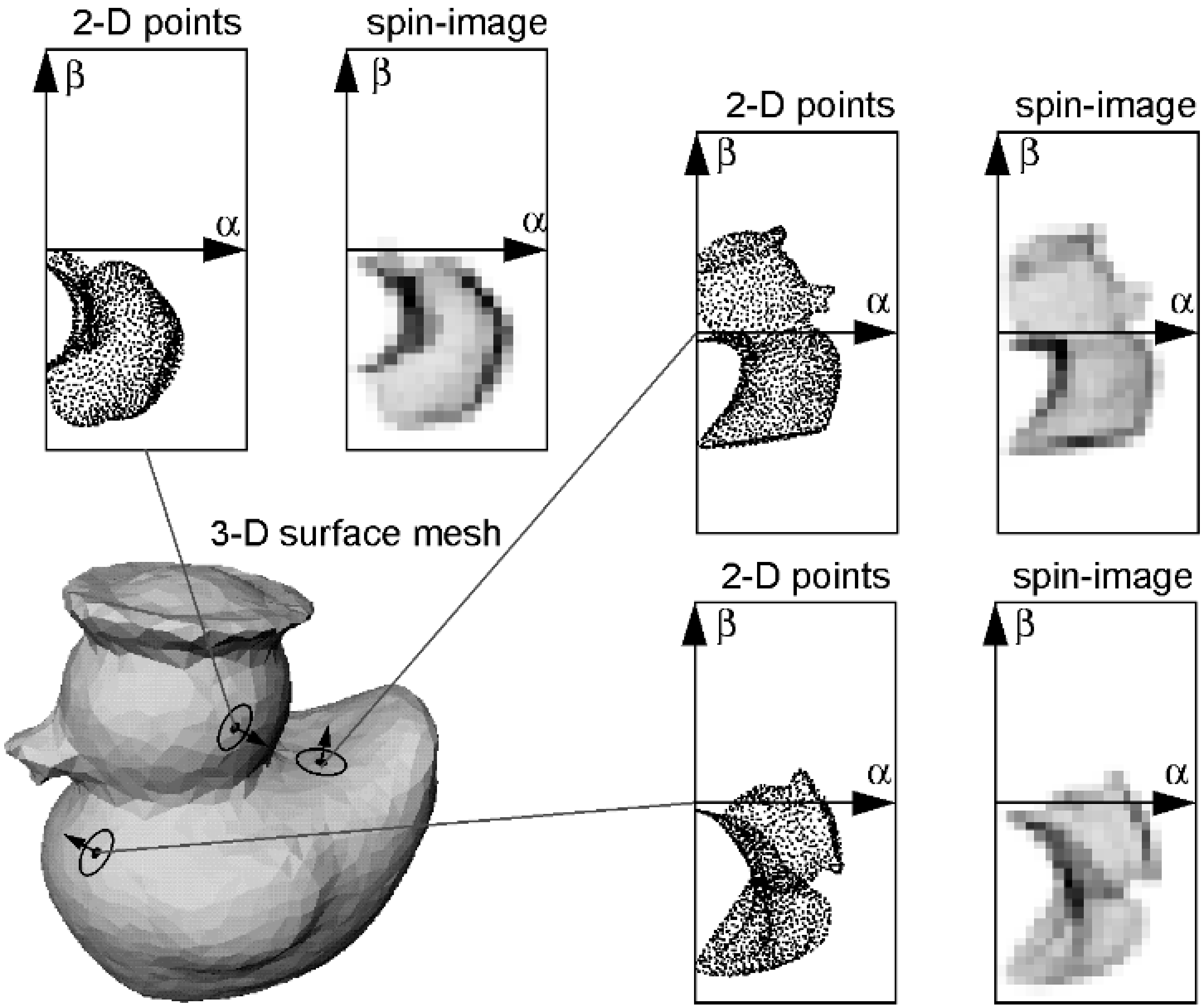
- **Spin Images (1997)**: Store energy in each normal ring



Spin images

- average of some geometry info, such as surface area, number of vertex, by intersecting the local geometry with rings => 2D histogram



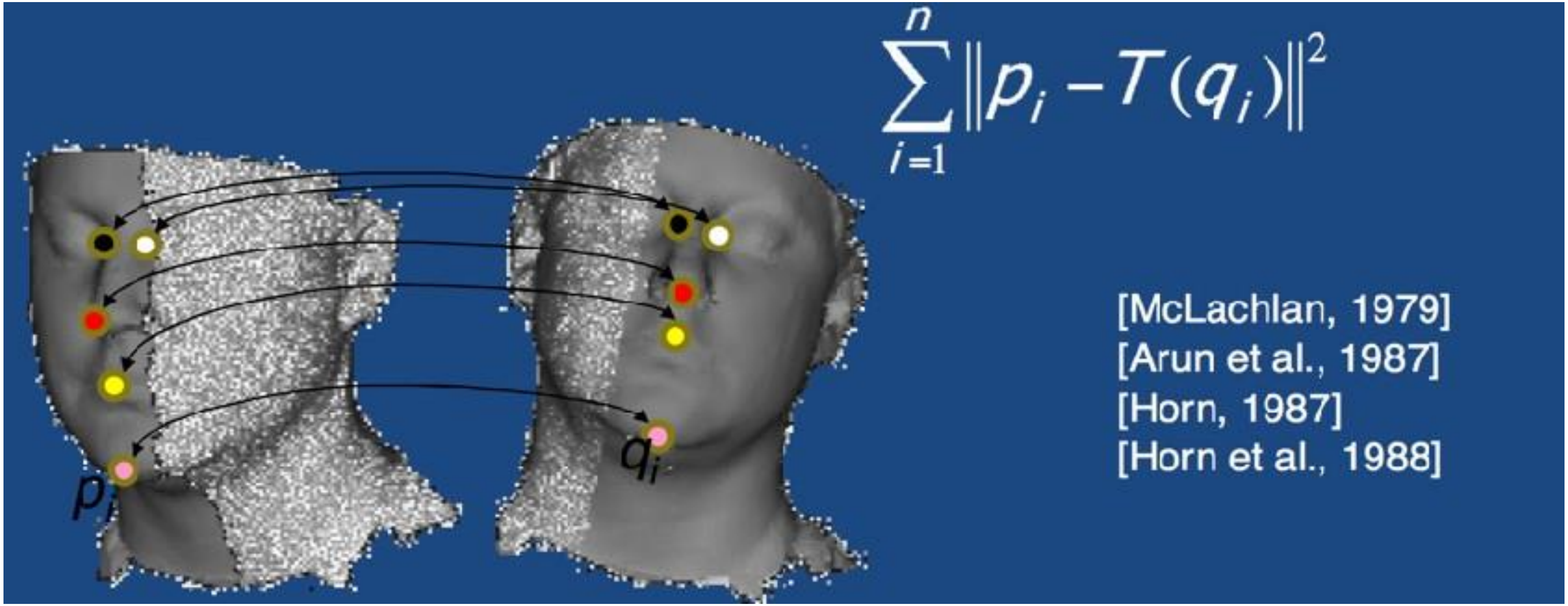


Outline

- Global Shape Correspondence
  - Shape Descriptors
  - Alignment
- Partial Shape Correspondence
  - From Global to Local
  - Pose Normalization
  - Partial Shape Descriptors
- Registration
  - Closed Form Solutions
  - Branch & Bound
  - Random Sample Consensus (RANSAC)

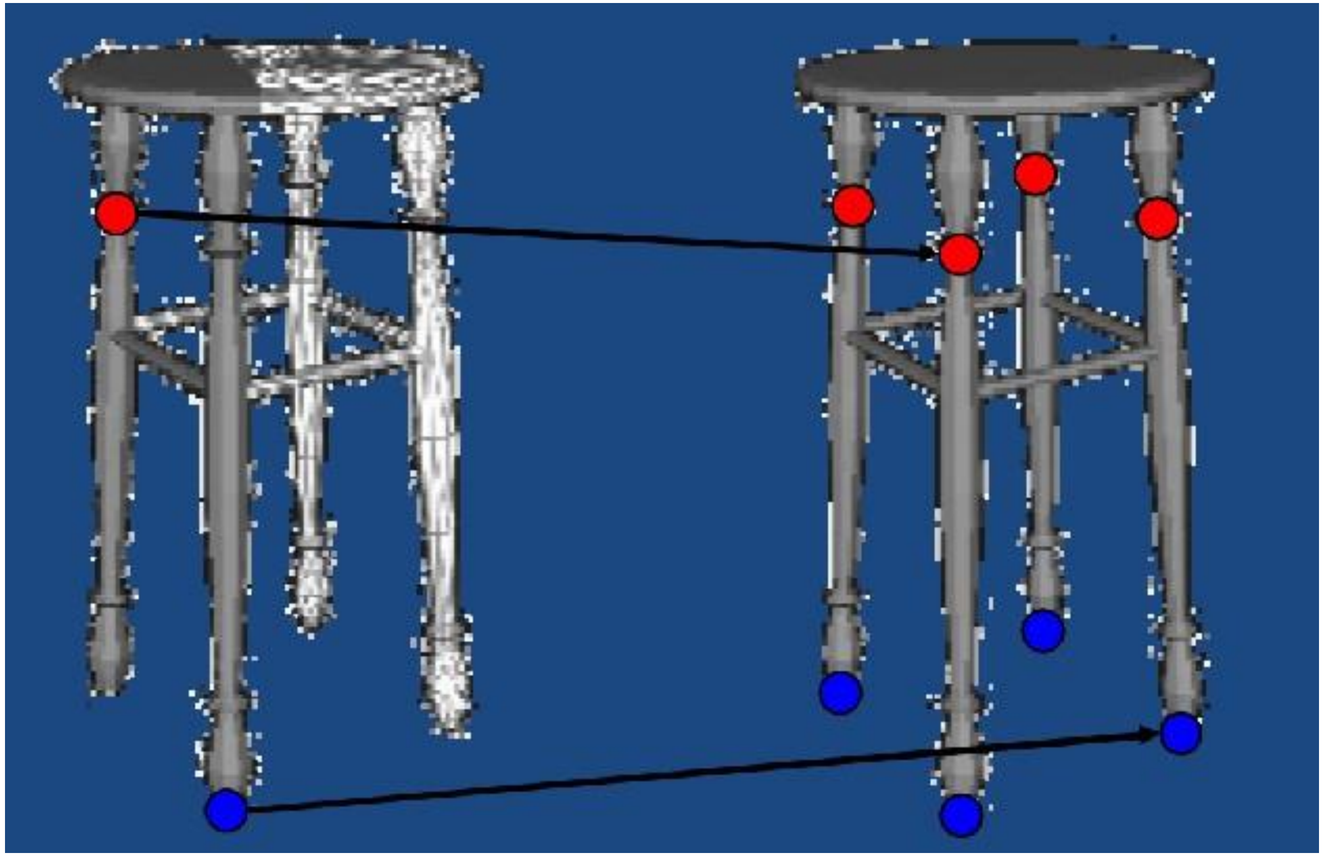
Registration - Ideal Case

- Every feature point on one scan has a single corresponding feature on the other.
- Solve for optimal transformation T



Registration - Challenge

- Even with good descriptors, **symmetries** in the model and the **locality** of descriptors can result in multiple and incorrect correspondences



we can no longer treat the correspondences independently!

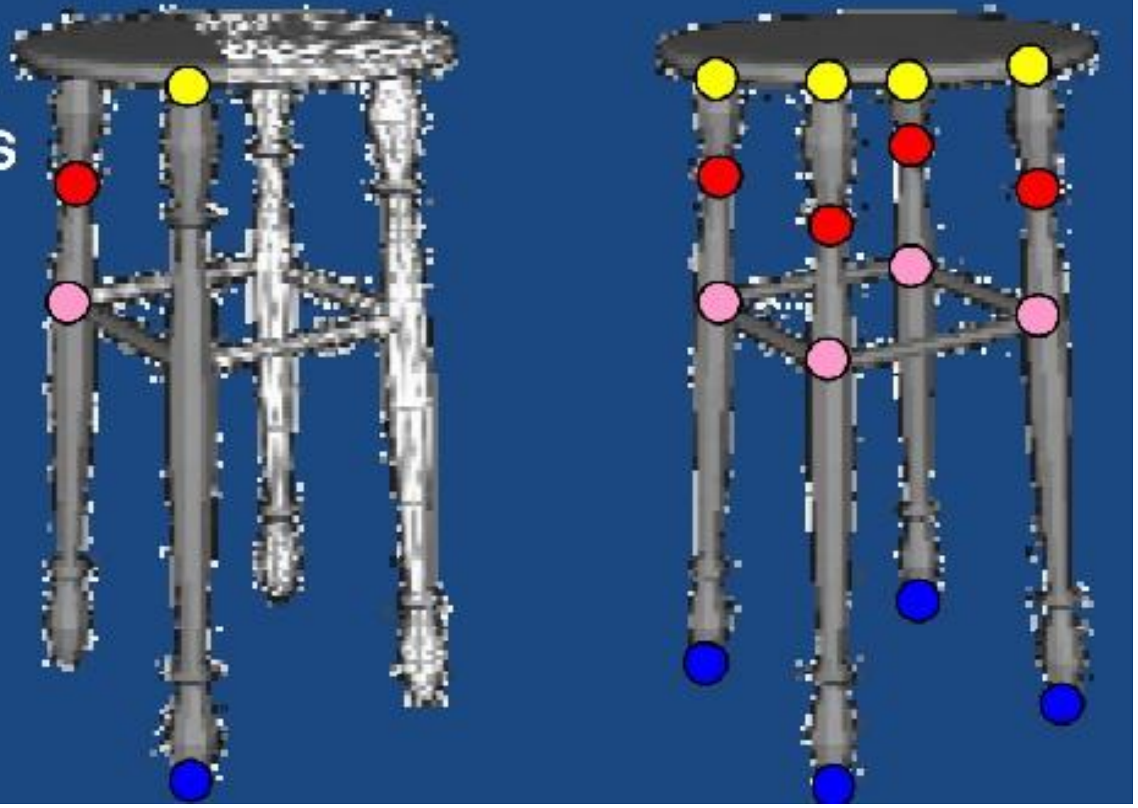
Registration - Exhaustive Search

- Compute alignment error at each permutation of correspondences and use the optimal one

$$\text{Error} = \underset{\pi \in \Psi}{\operatorname{argmin}} \left(\underset{T \in E^3}{\operatorname{argmin}} \sum_{i=1}^{n} \left\| p_i - T(\pi(p_i)) \right\|^2 \right)$$

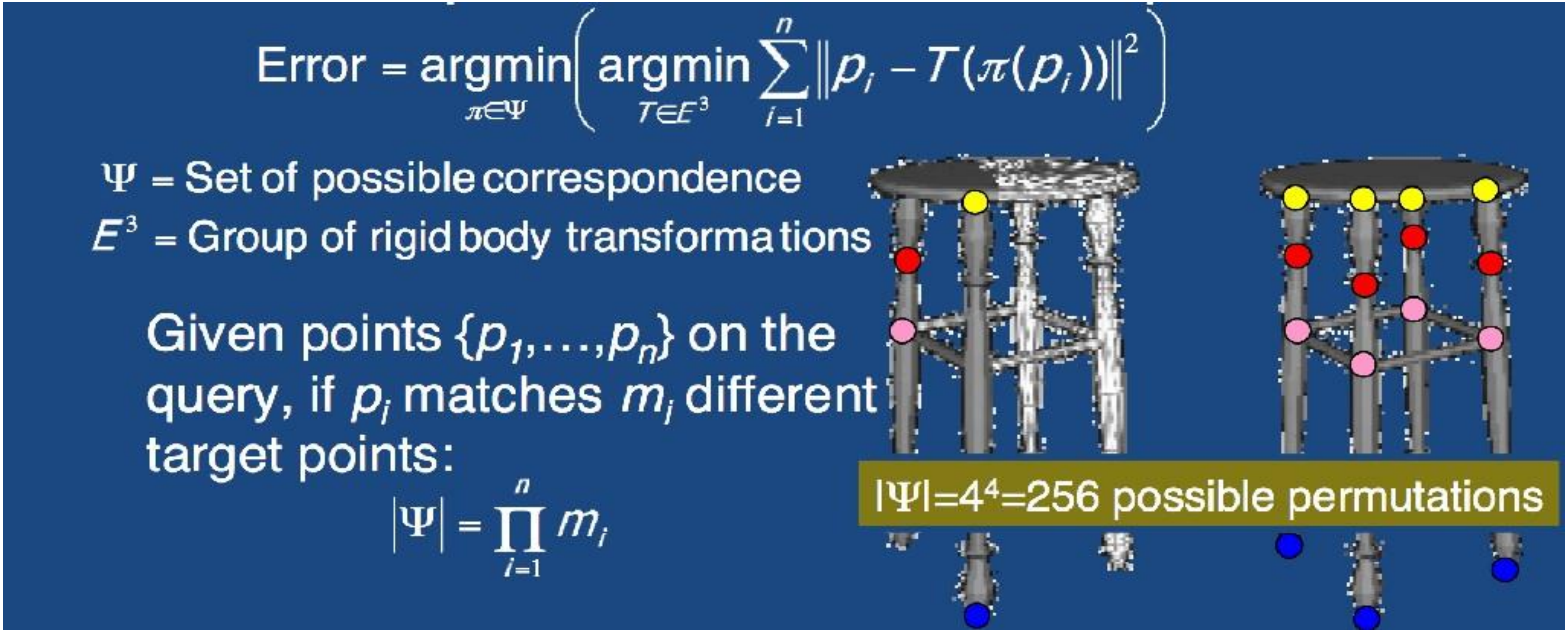
Ψ = Set of possible correspondence

E^3 = Group of rigid body transformations



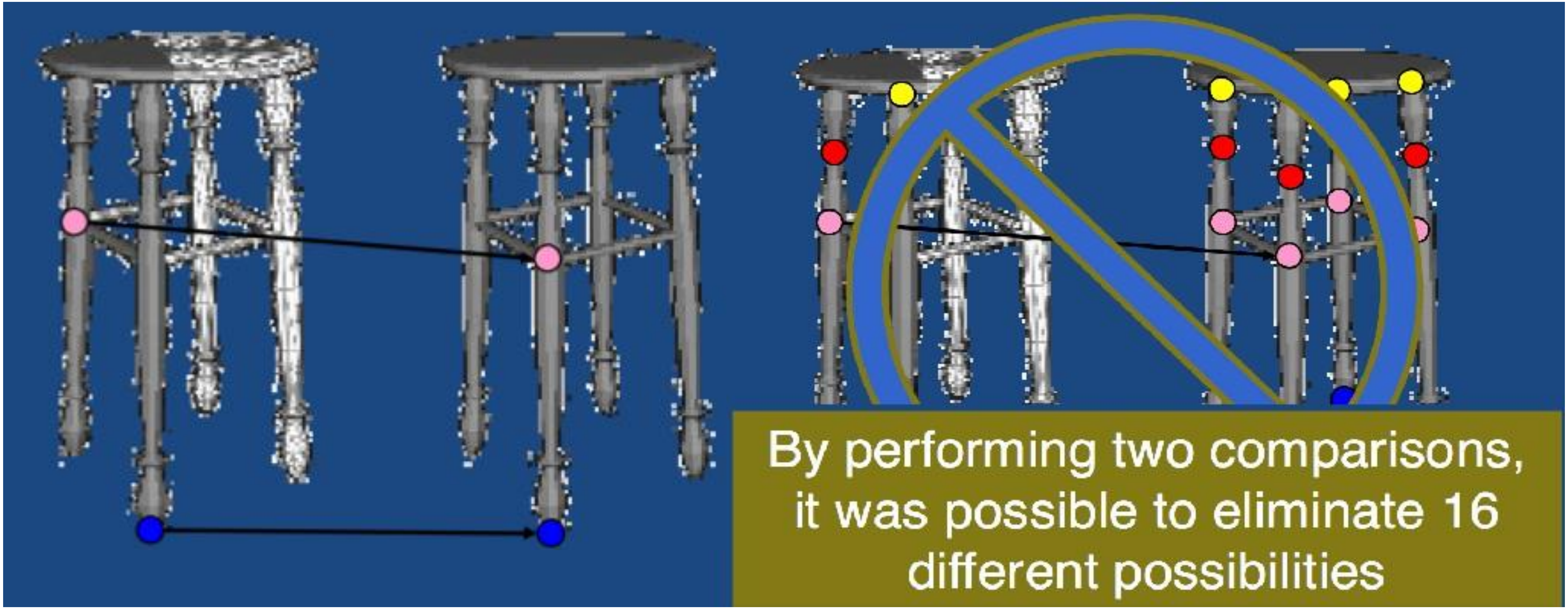
Registration - Exhaustive Search

- Compute alignment error at each permutation of correspondences and use the optimal one

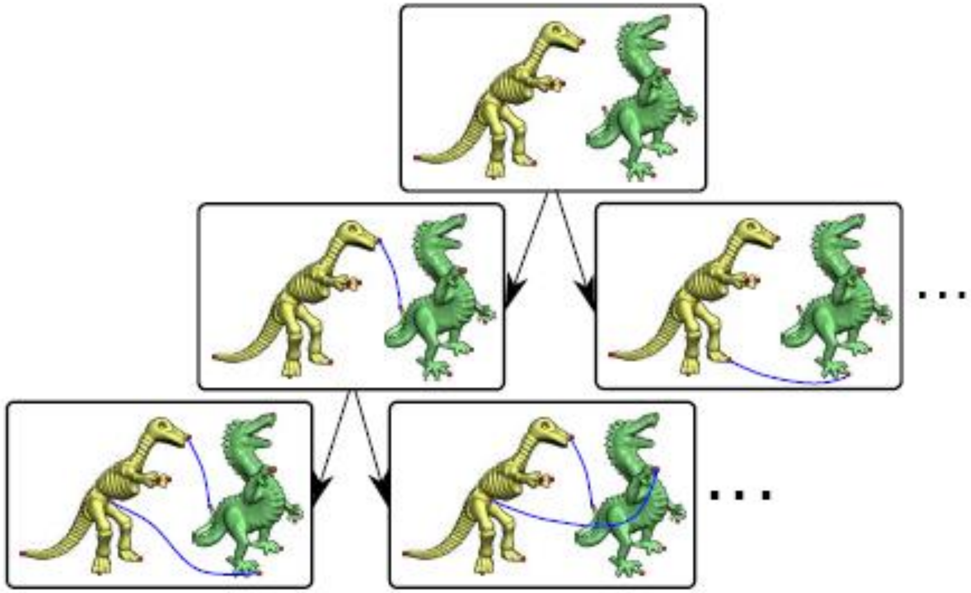


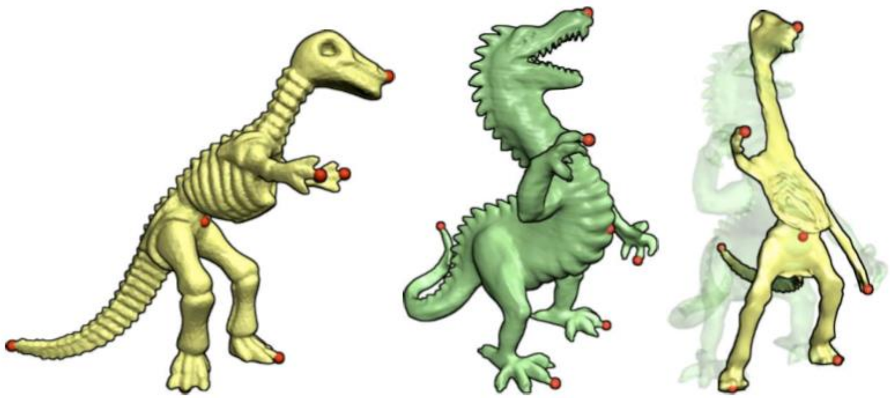
Registration - Branch & Bound (Decision tree)

- Try all permutations but **terminate early** if the alignment can be predicted to be bad

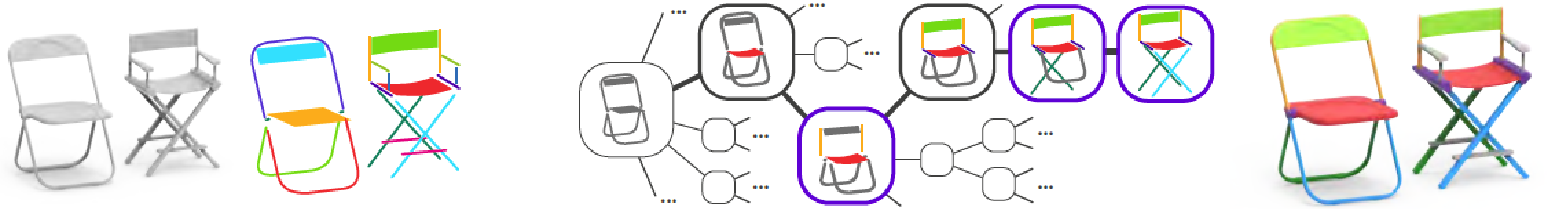


Tree-based search Branch & Bound





Cgf08-Deformation-Driven Shape Correspondence



(a) Source (left), target shapes, and curve-sheet abstractions (b) Search tree (c) Final correspondence result

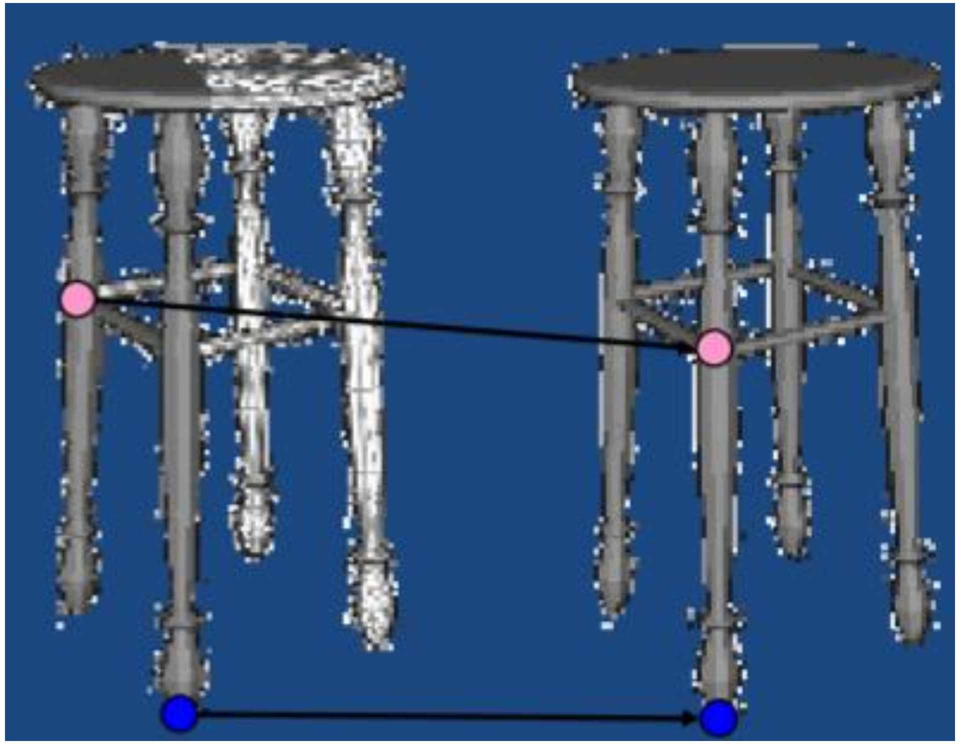
Siga15-Deformation-Driven Topology-Varying 3D Shape Correspondence

There are more papers using this in CG

Tree-based search Branch & Bound

Registration - Goal

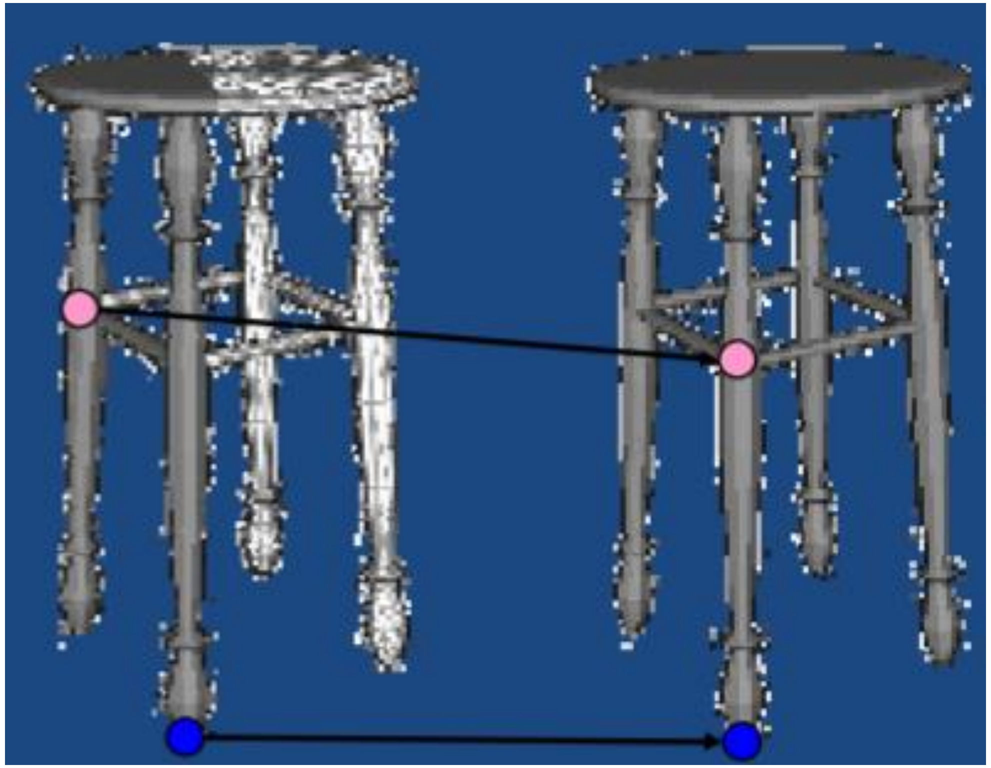
- Need to be able to determine if the alignment will be **good without knowing** all of the correspondences

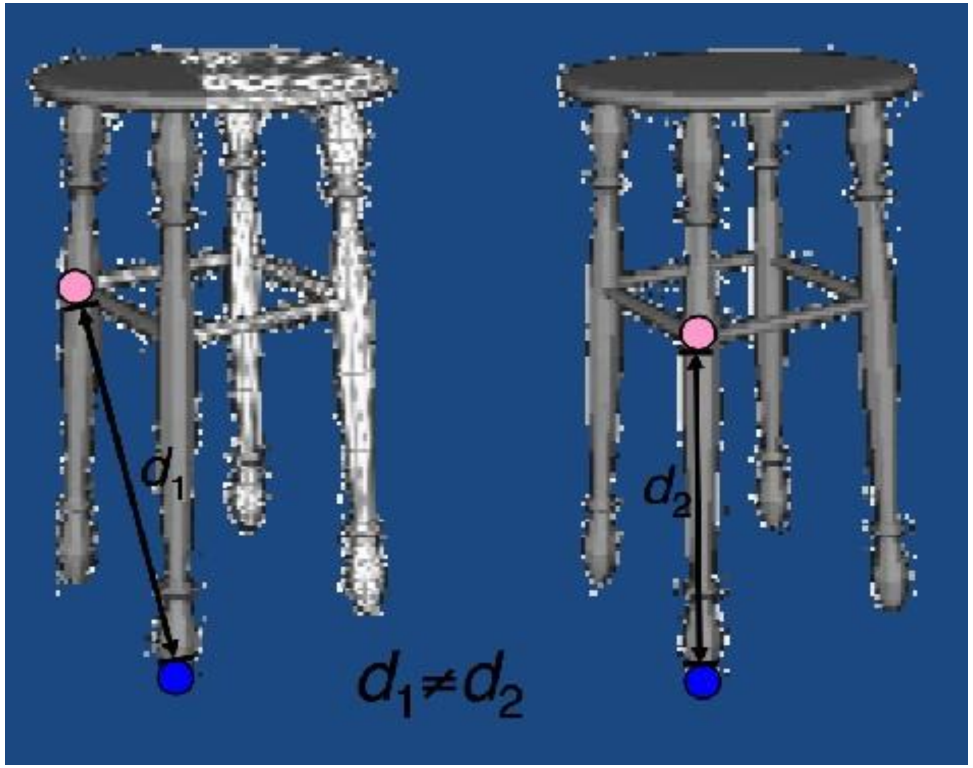


How to evaluate the alignment by using no more correspondences, i.e. just the two?

Registration - Goal

- Need to be able to determine if the alignment will be good without knowing all of the correspondences
- **Observation**
  - Alignment needs to preserve the lengths between points in a single scan

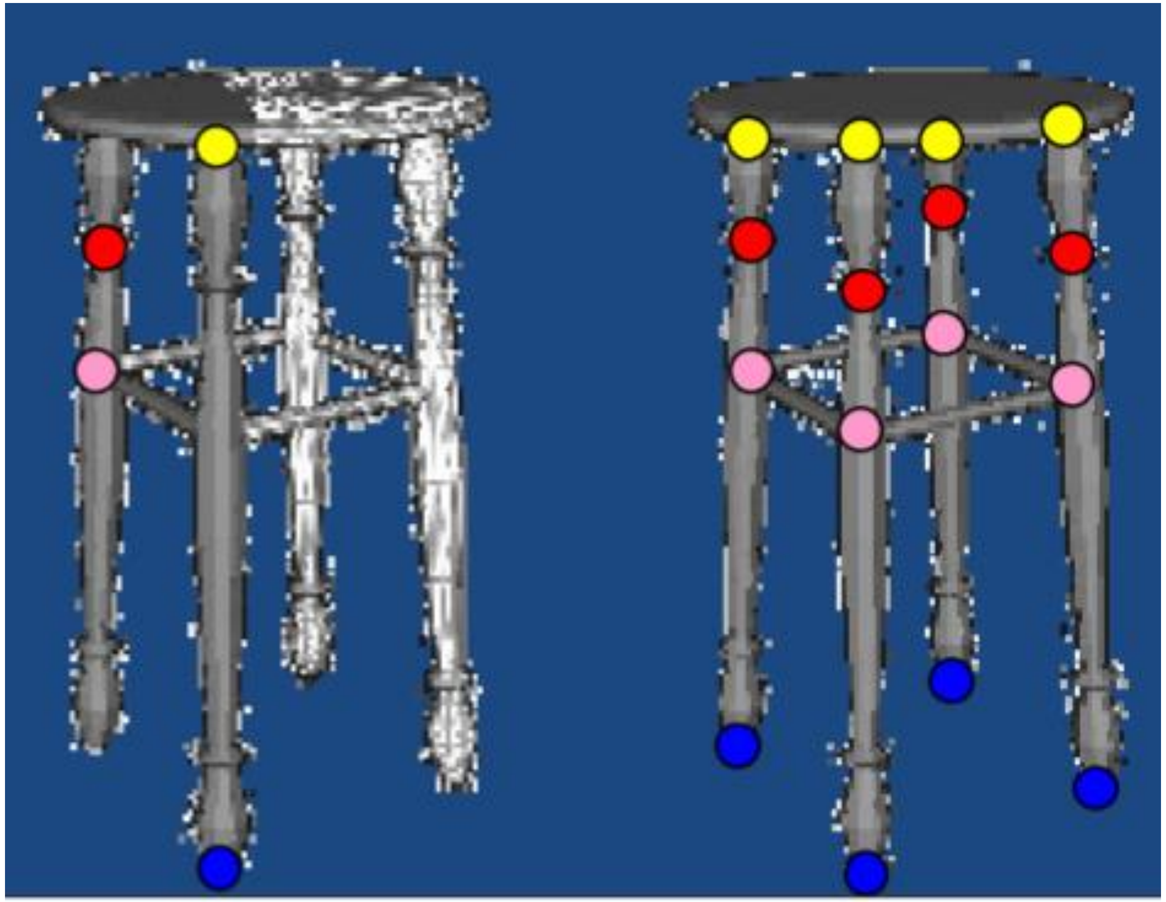




These distances **depend only on the correspondences and not on the alignment**

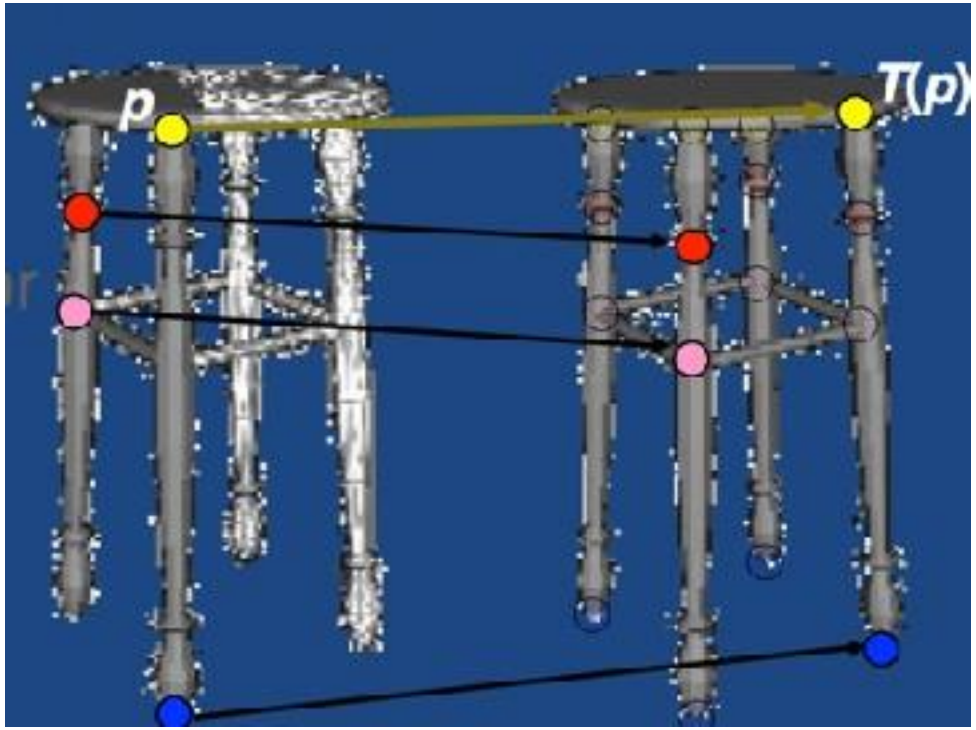
An alternative approach - **RANdom SAmple Consensus**

- Observation:
  - In 3D only three pairs of corresponding points are needed to define a transformation.



RANdom SAmple Consensus

- **Algorithm (iterate 100 times)**
- Randomly choose 3 points on source
- For all possible correspondences on target:
  - Compute T
  - For every other source p:
    - find closest correspondence T(p)
  - Compute alignment error



Summary

Global Shape Correspondences

- Shape Descriptors
  - Shells (1D)
  - Sectors (2D)
  - Sectors & Shells (3D)
- Alignment
  - Exhaustive Search
  - Normalization
  - Invariance

Summary

Partial-Shape/Point Correspondences

- From Global to Local
  - Center at feature
  - Restrict extent
- Pose Normalization
  - Normal-based alignment
- Partial Shape Descriptors
  - Normalization/invariance
  - Normalization/exhaustive-search

Summary

Registration

- Closed Form Solutions
  - Global symmetry
  - Local self similarity
- Branch & Bound
  - Inter-feature distances for early termination
- RANdom SAmple Consensus
  - Efficient transformation computation

Thanks