Shape Matching & Correspondence

CS 468 Geometry Processing Algorithms

Maks Ovsjanikov

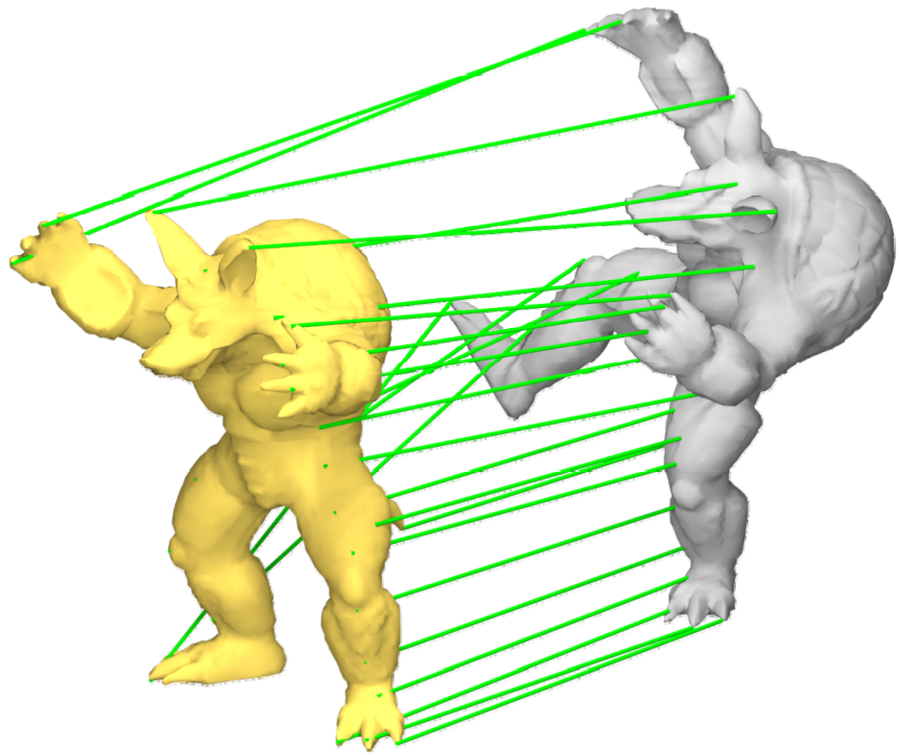
Wednesday, October 27th 2010

Overall Goal

- Given two shapes, find **correspondences** between them.

Overall Goal

- Given two shapes, find **correspondences** between them.



- Finding the **best** map between a pair of shapes.

Applications

- Manufacturing:

  One shape is a **model** and the other is a **scan** of a product. Finding defects.

- Medicine:

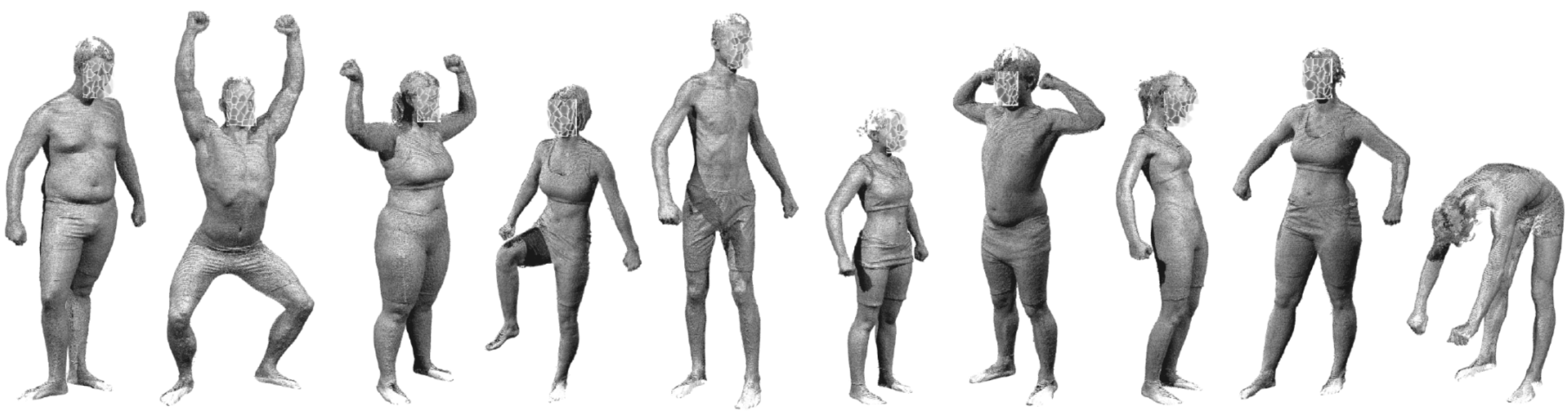
  Finding correspondences between 3D MRI scans of the same person or different people.

- Animation Reconstruction & 3D Video.

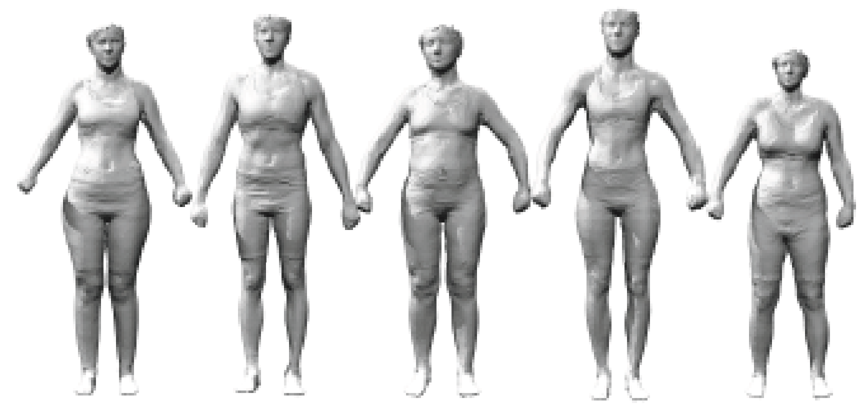
- Statistical Shape Analysis:

  Building models for a collection of shapes.

Applications – Statistical Analysis



- Scan many people. Learn the deformation model (e.g. PCA)
- Find the principal variation create new random instances.
- Requires correspondences.



female, 65kg

A Statistical Model of Human Pose and Body Shape ,
Hasler et al. Eurographics 09

Method Taxonomy

Local vs. Global

refinement (e.g. ICP) | alignment (search)

Rigid vs. Deformable

rotation, translation | general deformation

Pair vs. Collection

two shapes | multiple shapes

Method Taxonomy

Local vs. Global

refinement (e.g. ICP) alignment (search)

Solved

Rigid vs. Deformable

rotation, translation general deformation

Pair vs. Collection

two shapes multiple shapes

Method Taxonomy

Local vs. Global

refinement (e.g. ICP) | alignment (search)

Rigid vs. Deformable

rotation, translation | general deformation

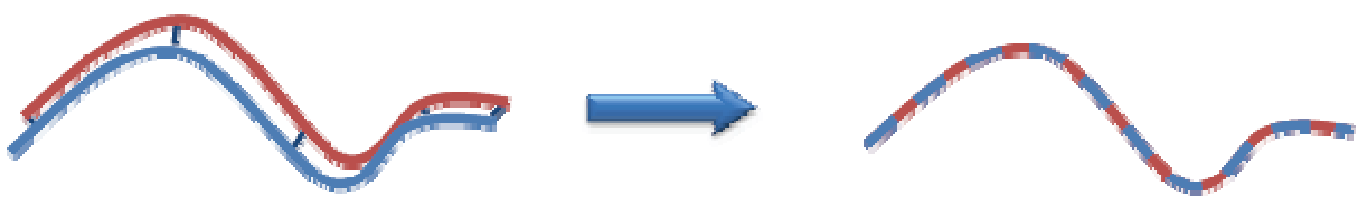
Pair vs. Collection

two shapes | multiple shapes

Most other combinations are open

Pairwise Rigid Correspondence

Remember ICP:



Given a pair of shapes, X and Y, iterate:

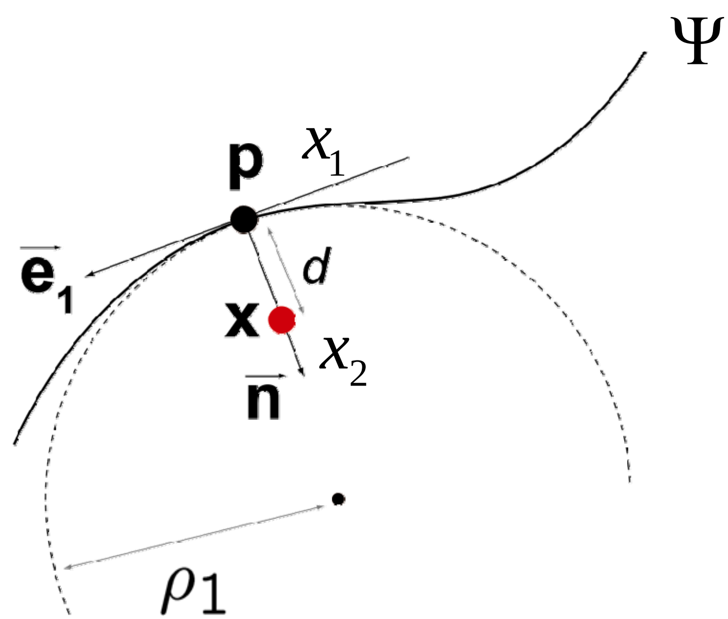
1. For each $x_i \in X$ find nearest neighbor $y_i \in Y$.
2. Find deformation $\mathbf{R}, t$ minimizing:

$$\sum_{x_i \in X} d^2(\mathbf{R}x_i + t, y_i)$$

Pairwise Rigid Correspondence

Geometry of the square distance function

For a curve Ψ,
around point x .



To second order:

$$d^2(y, \Psi) \approx \frac{d}{d-\rho_1} x_1^2 + x_2^2$$ in the Frenet frame at p

[Pottmann and Hofer 2003]

Approximate Squared Distance

For a curve Ψ, to second order:

$$d^2(y, \Psi) \approx \frac{d}{d - \rho_1} x_1^2 + x_2^2$$

For a surface Φ, to second order:

$$d^2(y, \Phi) \approx \frac{d}{d - \rho_1} x_1^2 + \frac{d}{d - \rho_2} x_2^2 + x_3^2$$

$\rho_1 = 1/\kappa_1$ and $\rho_2 = 1/\kappa_2$ are inverse principal curvatures

[Pottmann and Hofer 2003]

Approximate Squared Distance

For a surface Φ, to second order:

$$d^2(y,\Phi) \approx \frac{d}{d-\rho_1}x_1^2 + \frac{d}{d-\rho_2}x_2^2 + x_3^2$$

$\rho_1 = 1/\kappa_1$ and $\rho_2 = 1/\kappa_2$ are inverse principal curvatures

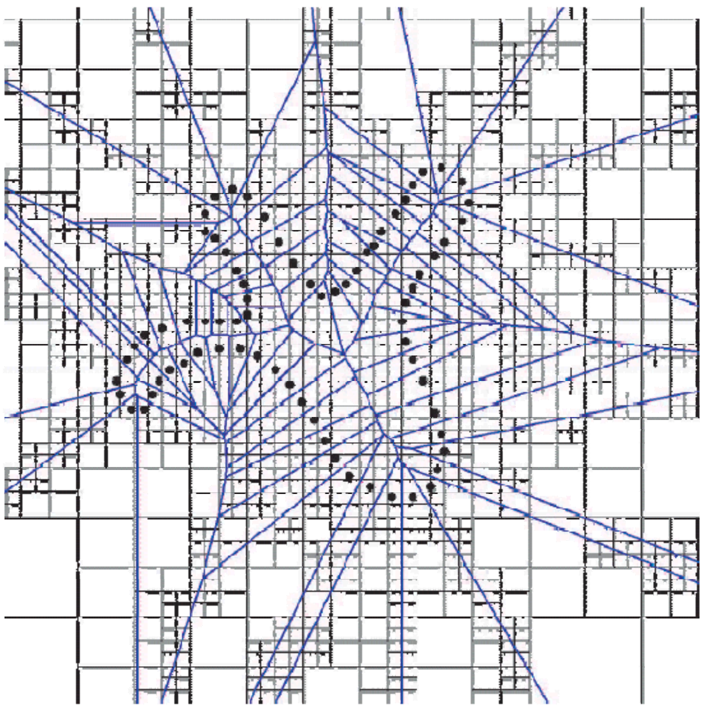
Note that as $d \to 0, \quad d^2(y,\Phi) \to x_3^2$ point-to-plane

$d \to \infty, \quad d^2(y,\Phi) \to x_1^2 + x_2^2 + x_3^2$ point-to-point

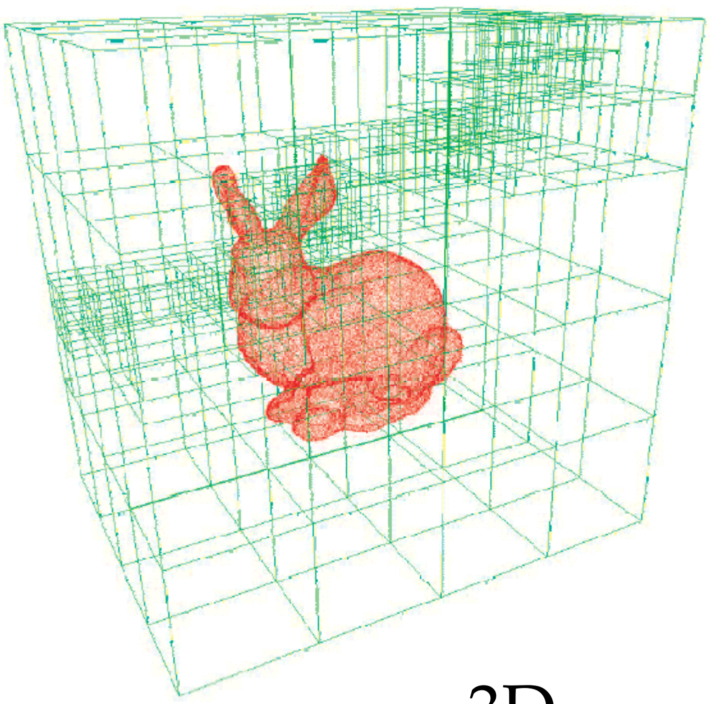
In general neither metric guarantees second order consistency

$d^2(y, \Phi_P)$ Using d2Tree

Partition the space into cells where each cell stores a quadratic approximant of the squared distance function.



2D



3D

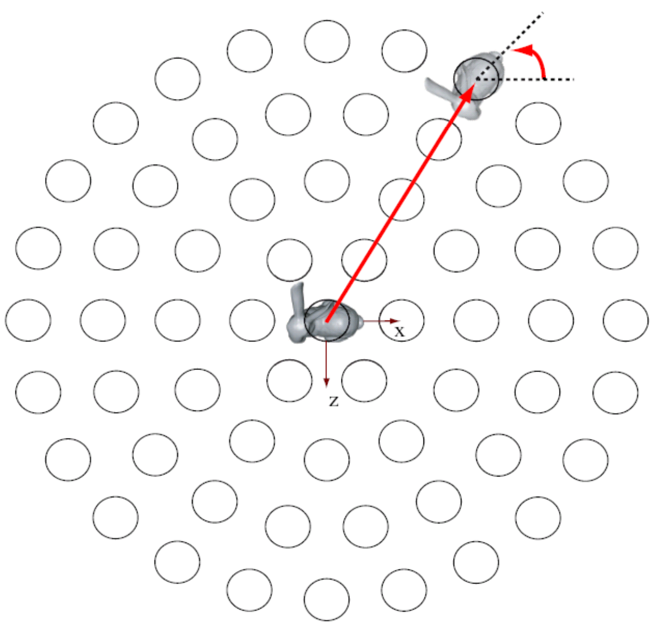
LEOPOLDSEDER S et al. *d2-tree: A hierarchical representation of the squared distance function*

Registration using d2Tree

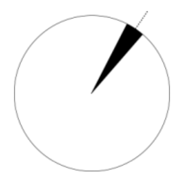
Build using bottom-up approach: fit a quadratic approximation to a fine grid.

Merge cells if they have similar approximations.

Funnel of Convergence:



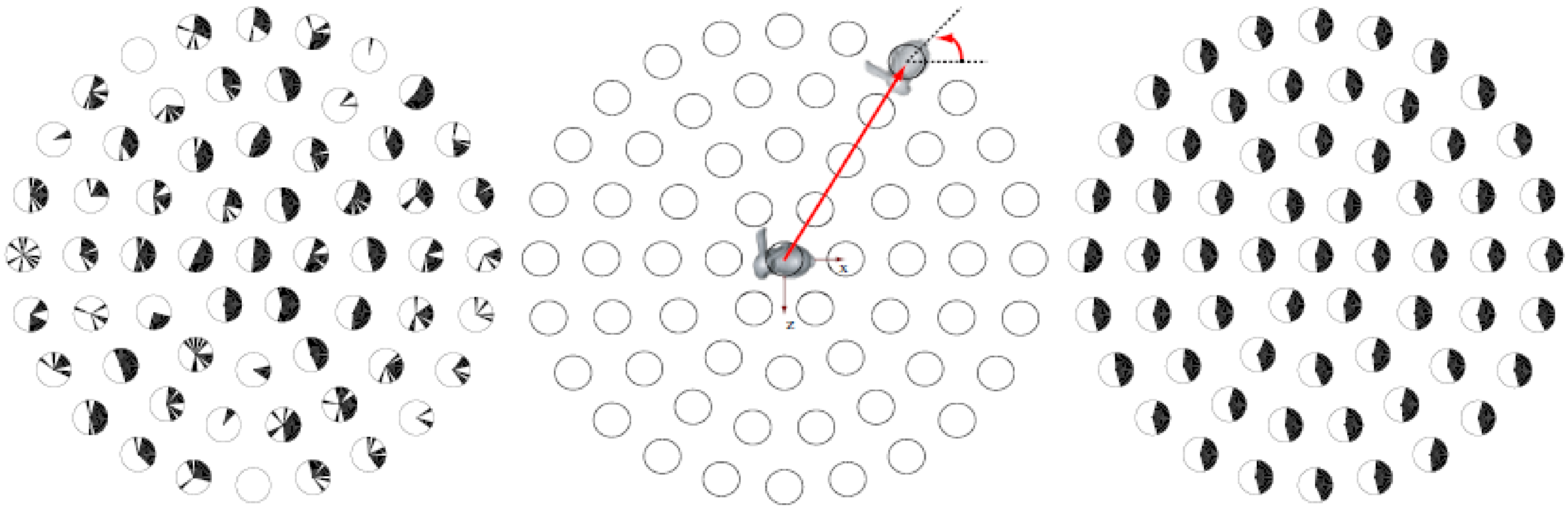
Translation in x-z plane.
Rotation about y-axis.



Converges

Does not converge

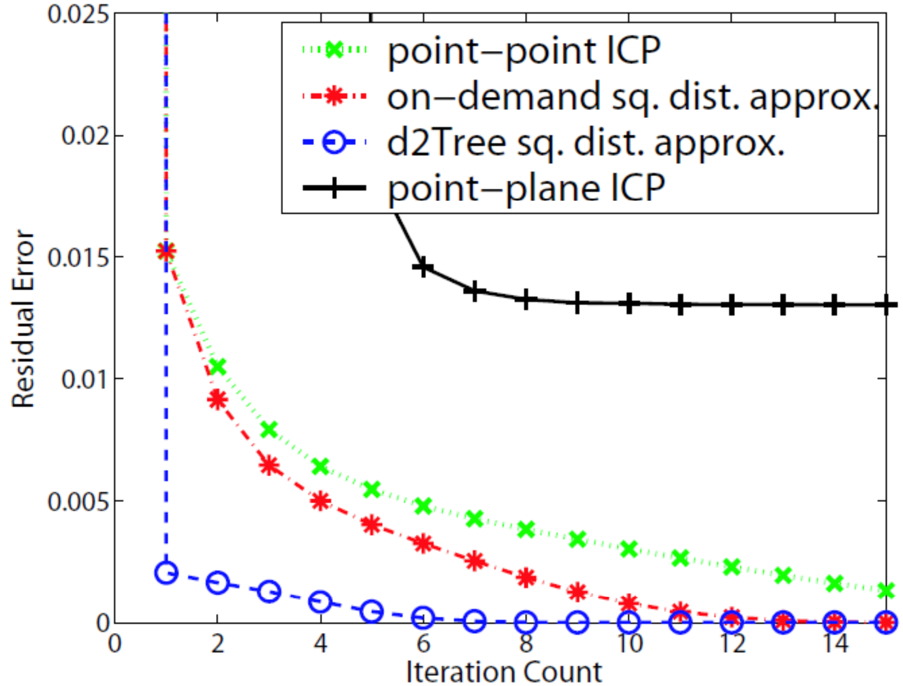
Matching the Bunny to itself



point-to-plane d2Tree

Registration of Point Cloud Data from a Geometric Optimization Perspective
Mitra et al., SGP 2004

Matching the Bunny to itself



Registration of Point Cloud Data from a Geometric Optimization Perspective
Mitra et al., SGP 2004

Local Rigid Matching – ICP

The upshot is that

- Locally, the point-to-plane metric provides a second order approximation to the squared distance function.
- Optimization based on point-to-plane will converge quadraticaly to the local minimum.
- Convergence funnel can be narrow, but can improve it with either d2tree or point-to-point.

What if we are outside the convergence funnel?

Global Matching

Given shapes in *arbitrary* positions, find their alignment:

Robust Global Registration
Gelfand et al. SGP 2005

Can be approximate, since will refine later using e.g. ICP

Global Matching – Approaches

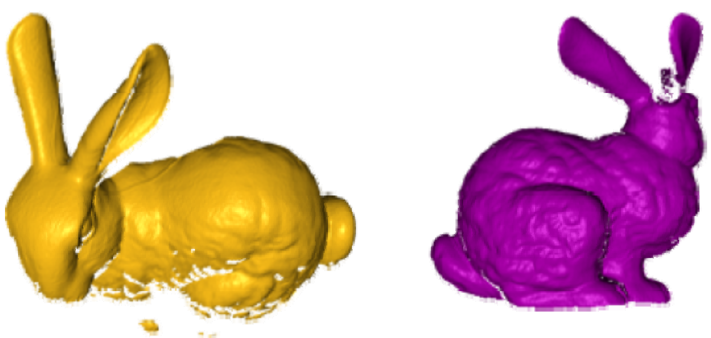
Several classes of approaches:

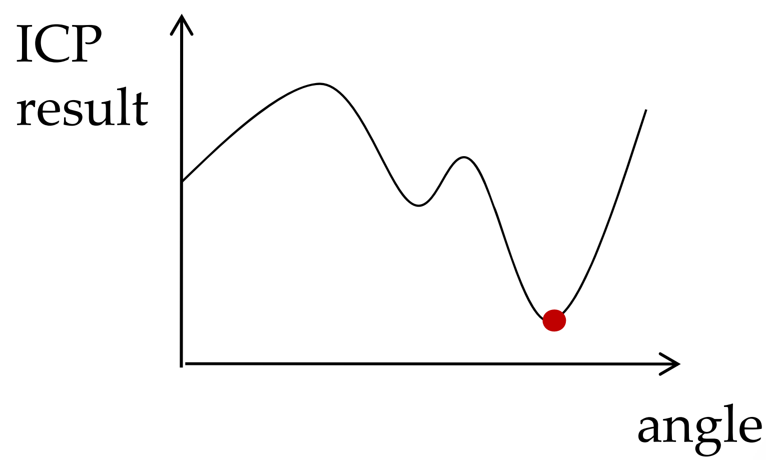
1. Exhaustive Search
2. Normalization
3. Random Sampling
4. Invariance

Exhaustive Search:

Compare at all alignments

- Sample the space of possible initial alignments.
- Correspondence is determined by the alignment at which models are closest.





Very common in biology: e.g., protein docking

Exhaustive Search:

Compare at all alignments

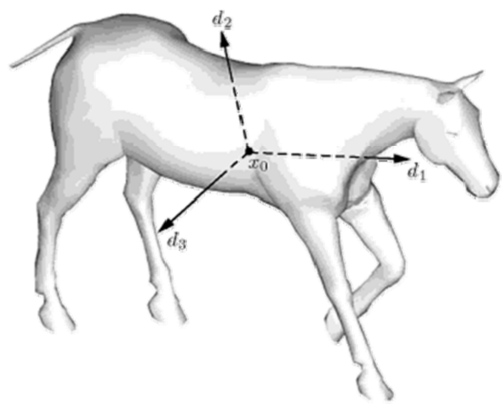
- Sample the space of possible initial alignments.
- Correspondence is determined by the alignment at which models are closest.

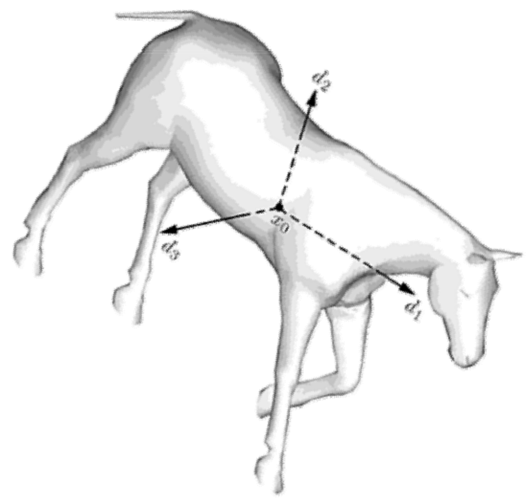
- Provides optimal result
- Can be unnecessarily slow.
- Does not generalize to non-rigid deformations.

Normalization

There are only a handful of initial configurations that are important.

Can center all shapes at the origin and use PCA to find the principal directions of the shape.

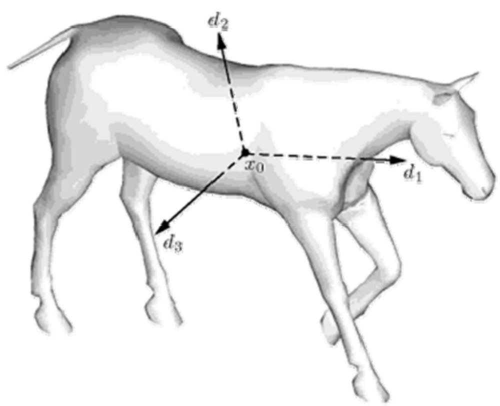




In addition sometimes try to all permutations of x-y-z.

Normalization

There are only a handful of initial configurations that are important.



Works well if we have complete shapes and no noise.

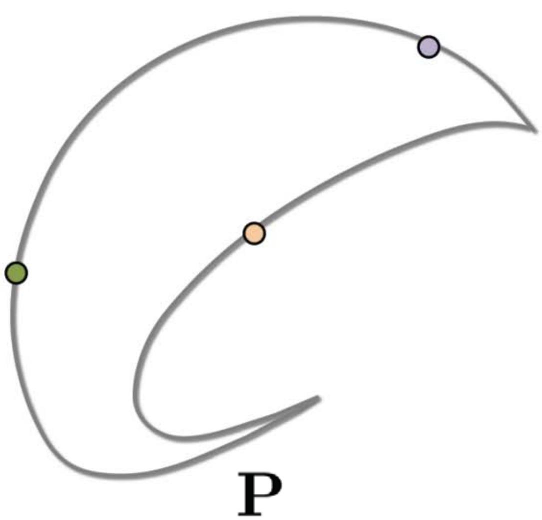
Fails for partial scans, outliers, high noise, etc.

Random Sampling (RANSAC)

ICP only needs 3 point pairs!

Robust and Simple approach. Iterate between:

1. Pick a random pair of 3 points on model & scan
2. Estimate alignment, and check for error.



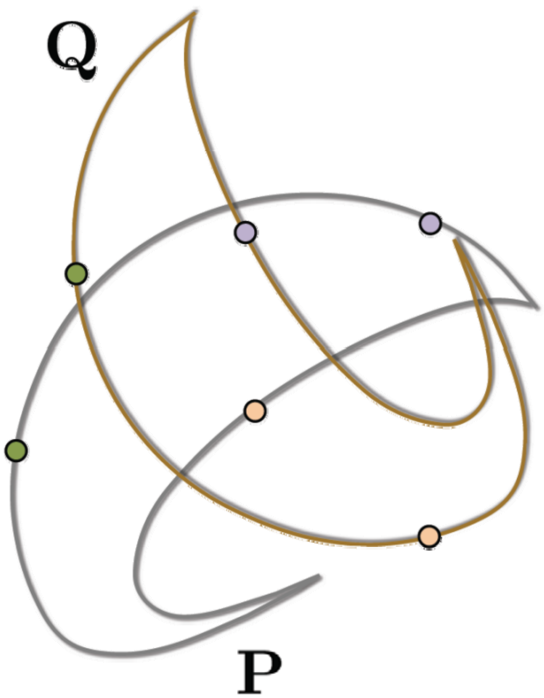
Guess and verify

Random Sampling (RANSAC)

ICP only needs 3 point pairs!

Robust and Simple approach. Iterate between:

1. Pick a random pair of 3 points on model & scan
2. Estimate alignment, and check for error.



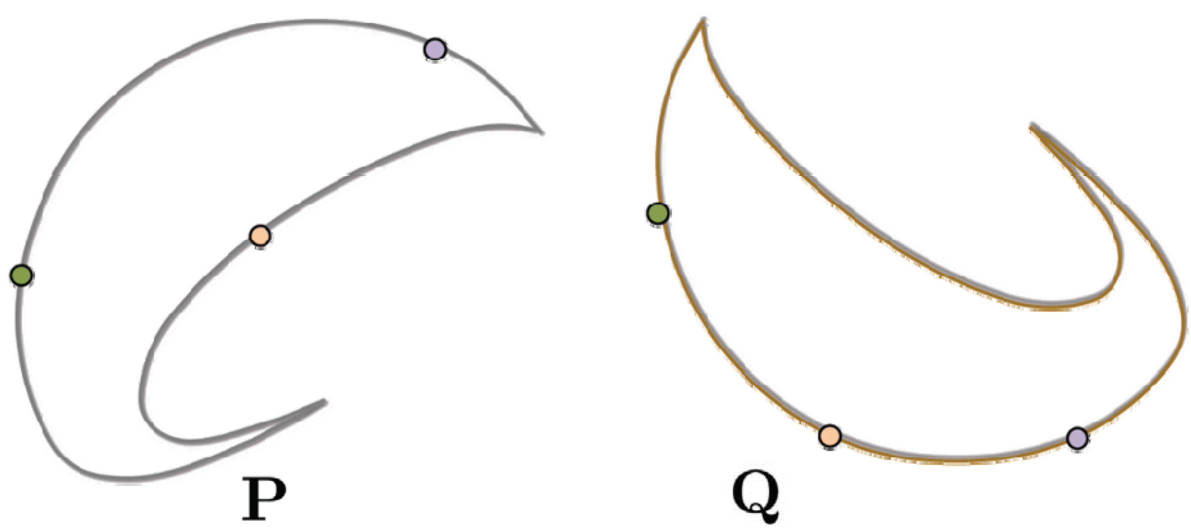
Guess and verify

Random Sampling (RANSAC)

ICP only needs 3 point pairs!

Robust and Simple approach. Iterate between:

1. Pick a random pair of 3 points on model & scan
2. Estimate alignment, and check for error.



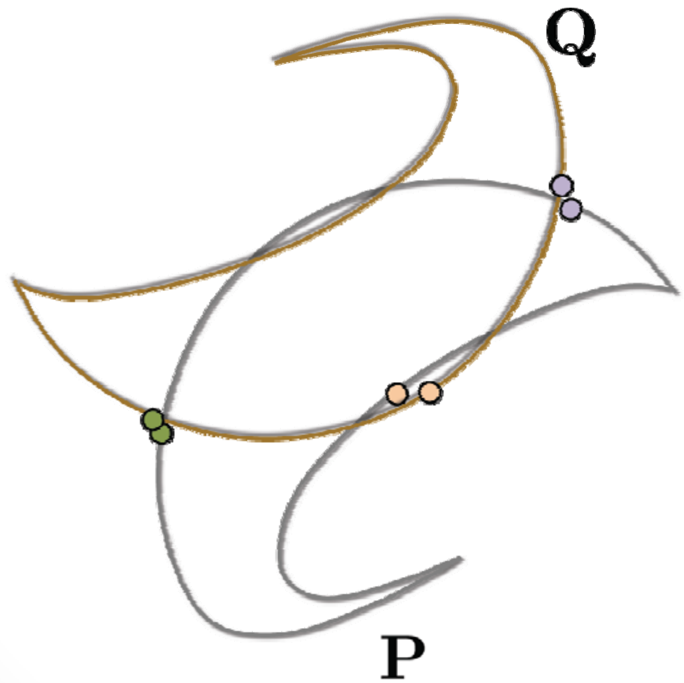
Guess and verify

Random Sampling (RANSAC)

ICP only needs 3 point pairs!

Robust and Simple approach. Iterate between:

1. Pick a random pair of 3 points on model & scan
2. Estimate alignment, and check for error.



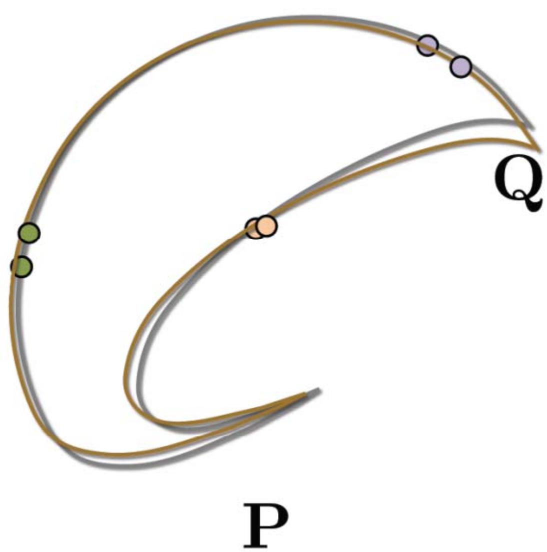
Guess and verify

Random Sampling (RANSAC)

ICP only needs 3 point pairs!

Robust and Simple approach. Iterate between:

1. Pick a random pair of 3 points on model & scan
2. Estimate alignment, and check for error.



Can also refine the final result. Picks don't have to be exact.

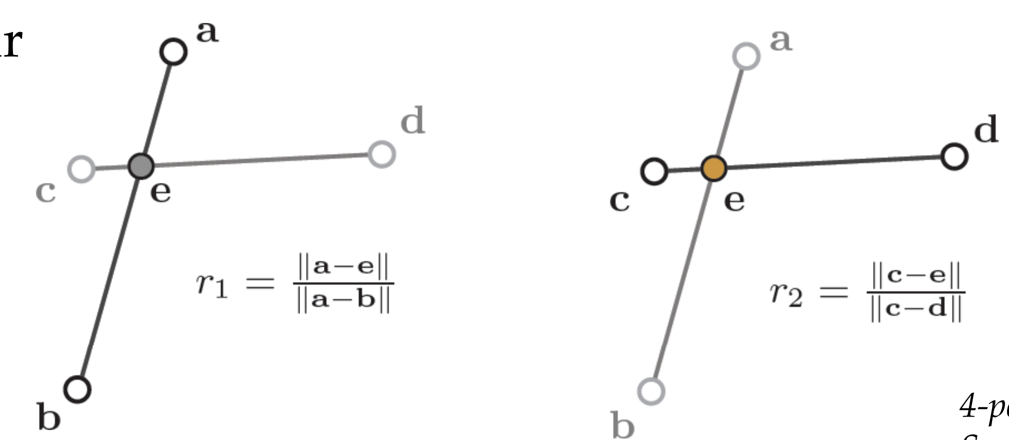
Guess and verify

Random Sampling (RANSAC)

A pair of triples (from **P** and **Q**) are enough to determine a **rigid transform**, resulting in $O(n^3)$ RANSAC.

Surprisingly, a special set of 4 points, **congruent sets**, makes the problem simpler leading to $O(n^2)$!

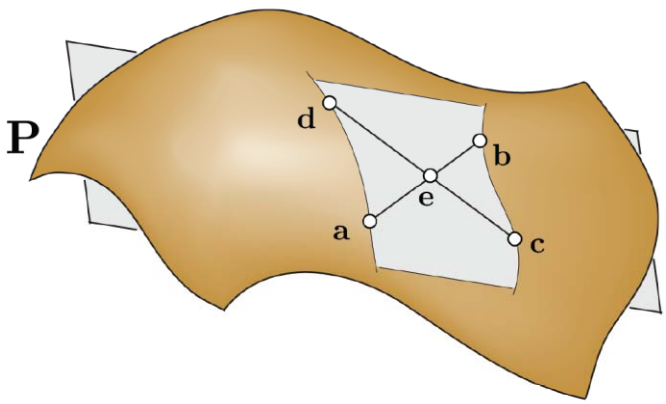
Co-planar points remain coplanar



4-points Congruent Sets for Robust Surface Registration,
Aiger et al., SIGGRAPH 2008

Method Overview

On the source shape, pick 4 (approx.) coplanar points.



Compute

$$r_1 = \frac{\|a - e\|}{\|a - b\|} \qquad r_2 = \frac{\|d - e\|}{\|d - c\|}$$

For every *pair* (q_1, q_2) of points on the destination compute

$$p_1 = q_1 + r_1(q_2 - q_1)$$

$$p_2 = q_1 + r_2(q_2 - q_1)$$

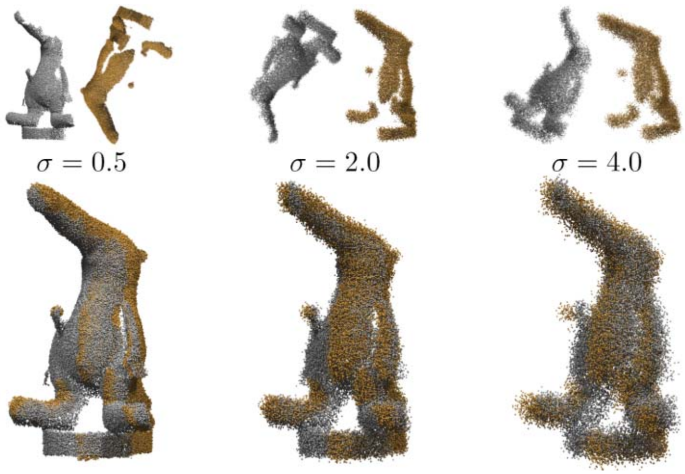
Those pairs (q_1, q_2) , (q_3, q_4) for which $p1_{(q_1,q_2)} = p2_{(q_3,q_4)}$ are a good candidate correspondence for (a, b, c, d).

Under mild assumptions the procedure runs in $O(n^2)$ time.

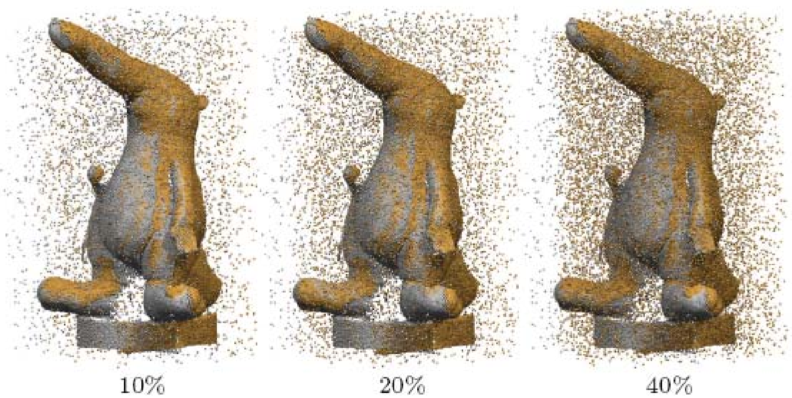
4-points Congruent Sets for Robust Surface Registration,
Aiger et al., SIGGRAPH 2008

Method Overview

Can pick a few base points for partial matching.



Random sampling
handles noise



and outliers

4-points Congruent Sets for Robust Surface Registration,
Aiger et al., SIGGRAPH 2008

Method Overview

Can pick a few base points for partial matching.

Partial matches

4-points Congruent Sets for Robust Surface Registration,
Aiger et al., SIGGRAPH 2008

Method Overview

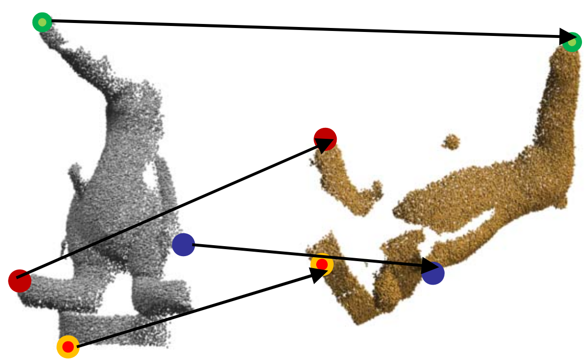
Very robust to noise, fast global alignment.

Can further improve efficiency.

Still need to local refinement.

Can further reduce the number of candidates. **Not all points are created equal**.

Find salient, feature points.



4-points Congruent Sets for Robust Surface Registration,
Aiger et al., SIGGRAPH 2008

Global Matching – Approaches

Several classes of approaches:

1. Exhaustive Search
2. Normalization
3. Random Sampling
4. Invariance

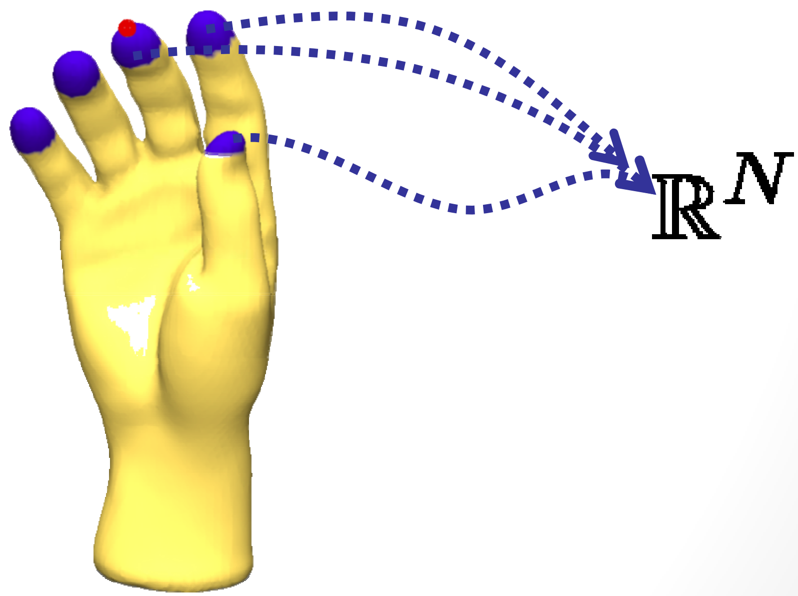
Global Matching – Invariance

Try to characterize the shape using properties that are invariant under the desired set of transformations.

Conflicting interests – invariance vs. informativeness.

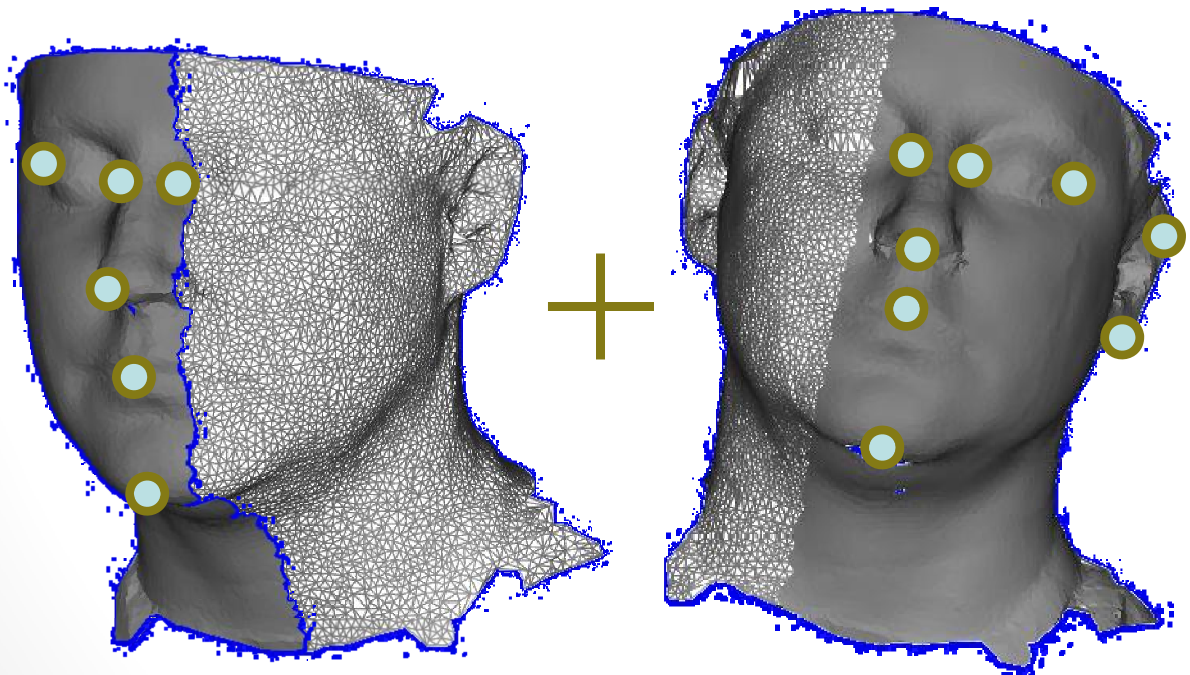
The most common pipeline:

1. identify salient feature points
2. compute informative and commensurable descriptors.



Matching Using Feature Points

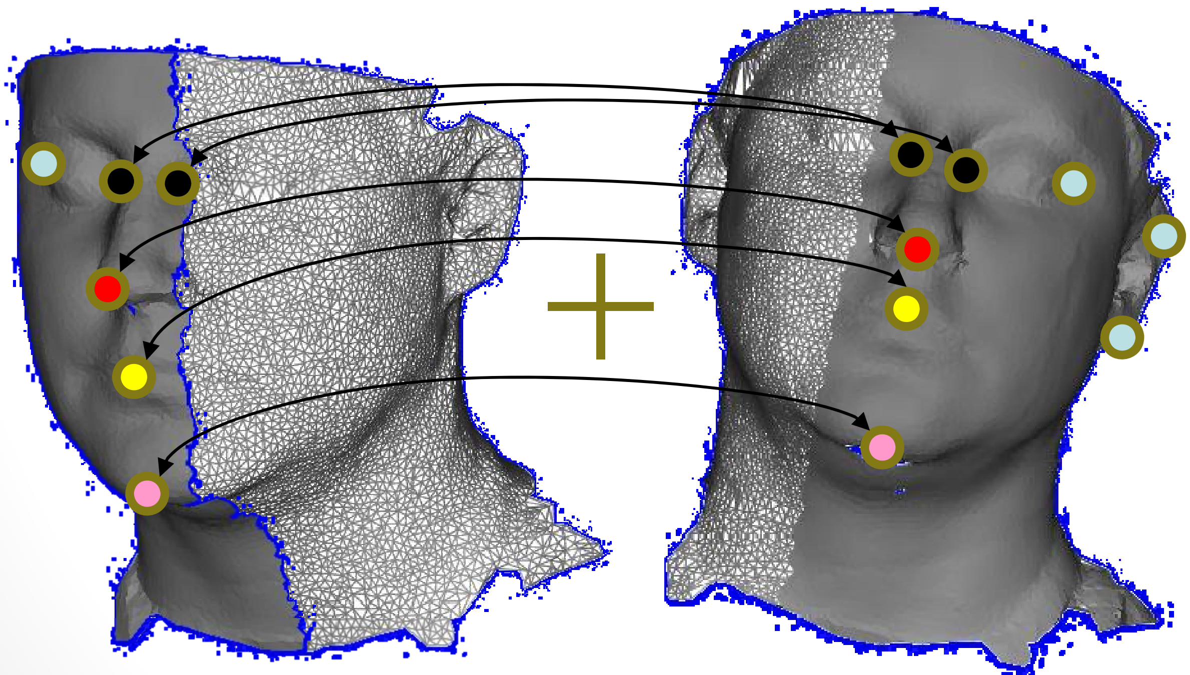
1. Find feature points on the two scans (we'll come back to that issue)



Partially Overlapping Scans

Approach

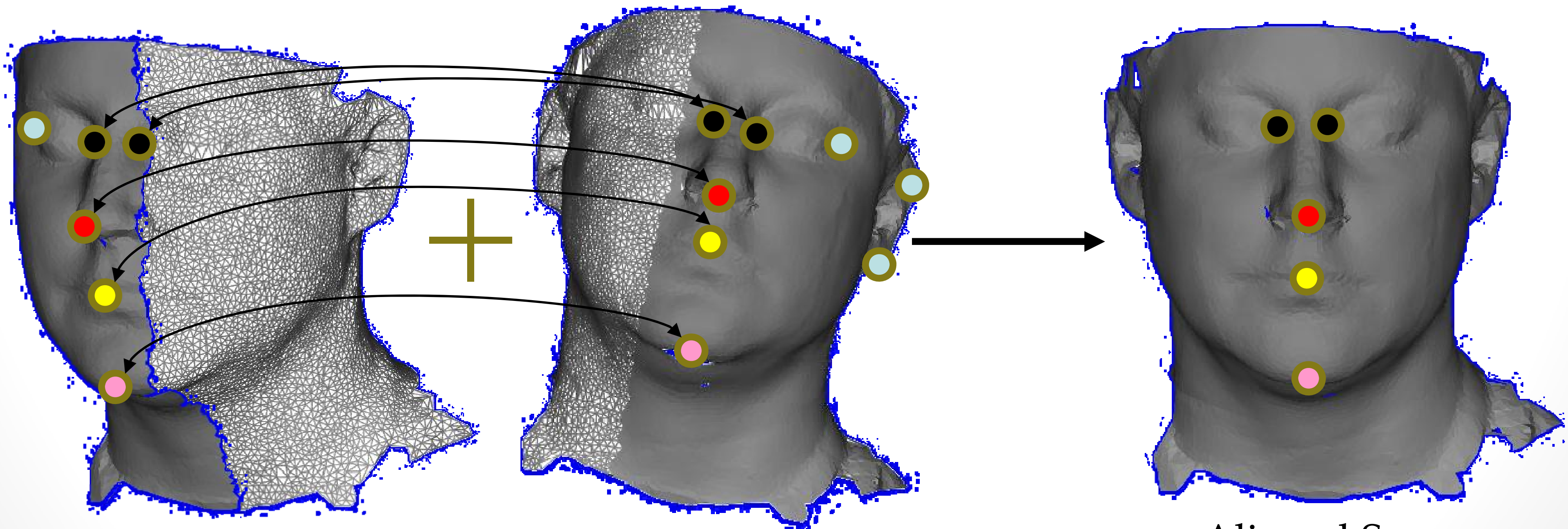
1. (Find feature points on the two scans)
2. Establish correspondences



Partially Overlapping Scans

Approach

1. (Find feature points on the two scans)
2. Establish correspondences
3. Compute the aligning transformation



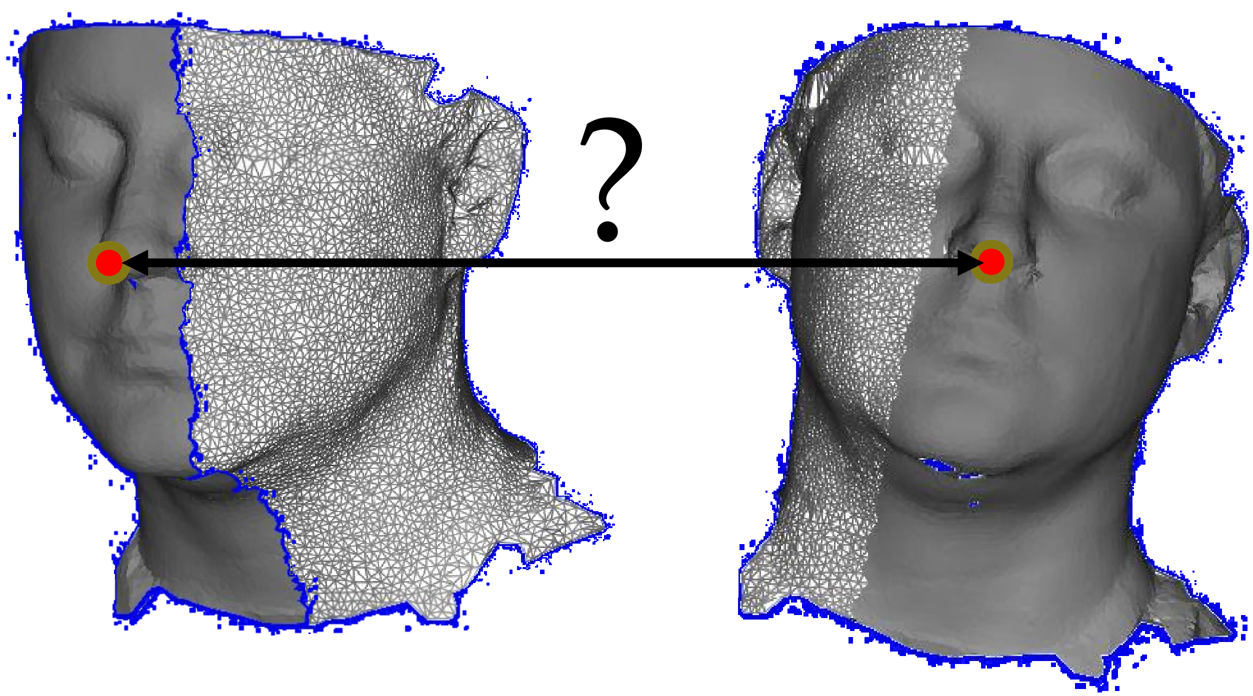
Partially Overlapping Scans

Aligned Scans

Correspondence

<u>Goal</u>:

Identify when two points on different scans represent the same feature

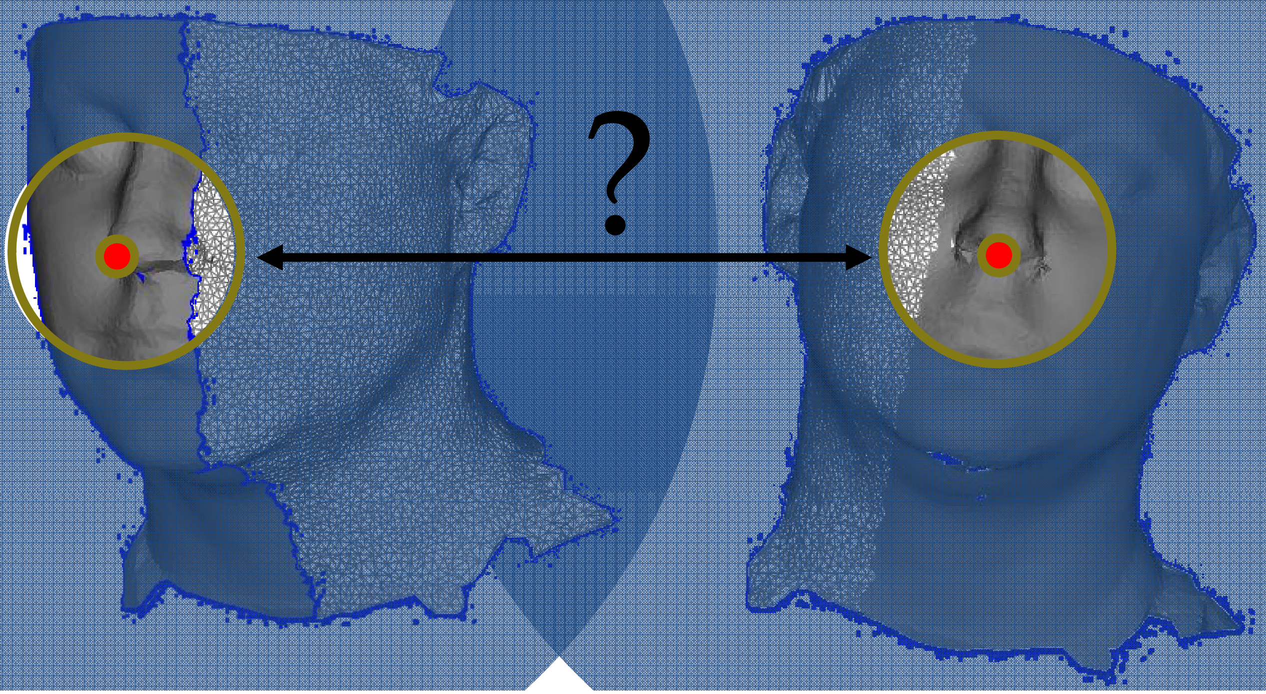


Correspondence

<u>Goal</u>:

Identify when two points on different scans represent the same feature:

Are the surrounding regions similar?

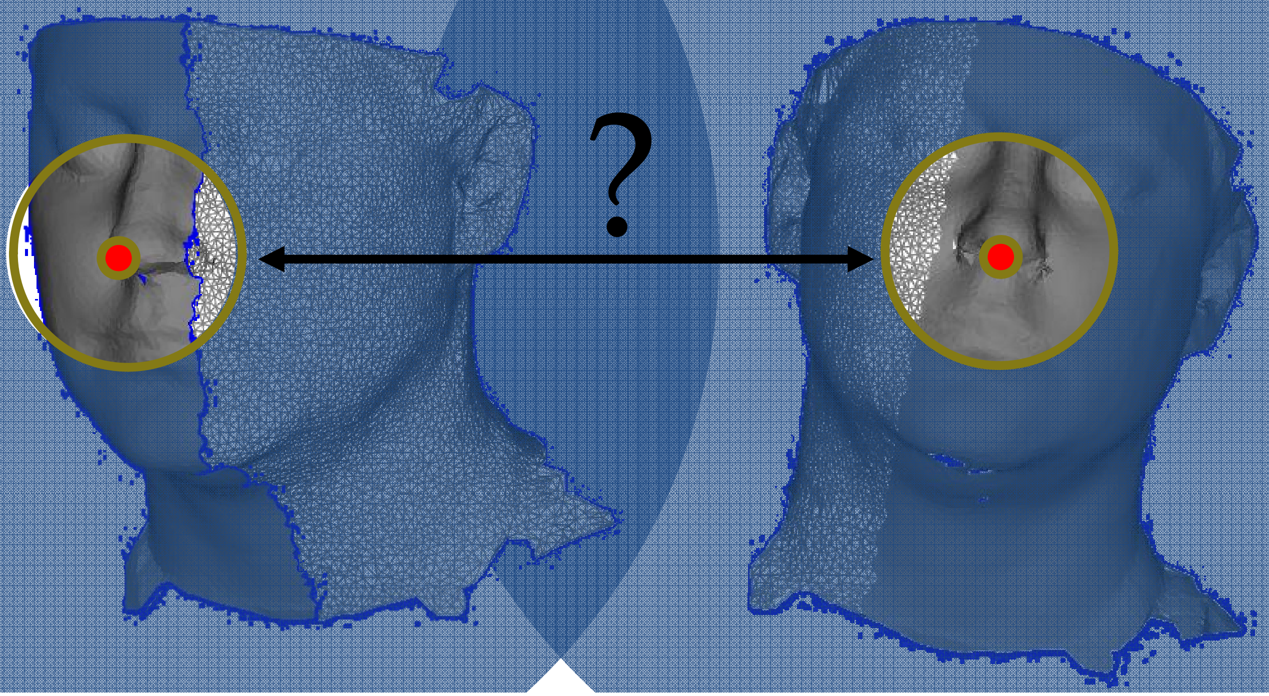


Correspondence

<u>Goal</u>:

Identify when two points on different scans represent the same feature:

Are the surrounding regions similar?



Main Question

How to compare regions on the shape in an invariant manner?

A large variety of *descriptors* have been suggested.

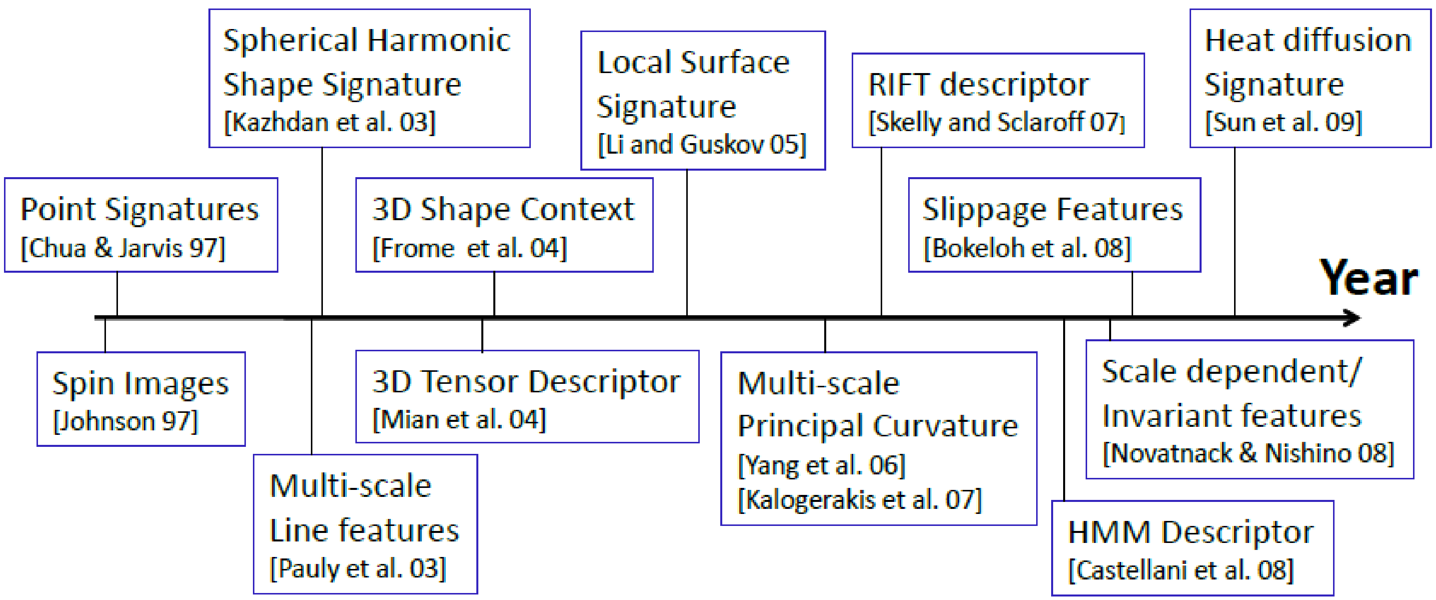


table by Will Chang

To give an example, I'll describe two.

Spin Images

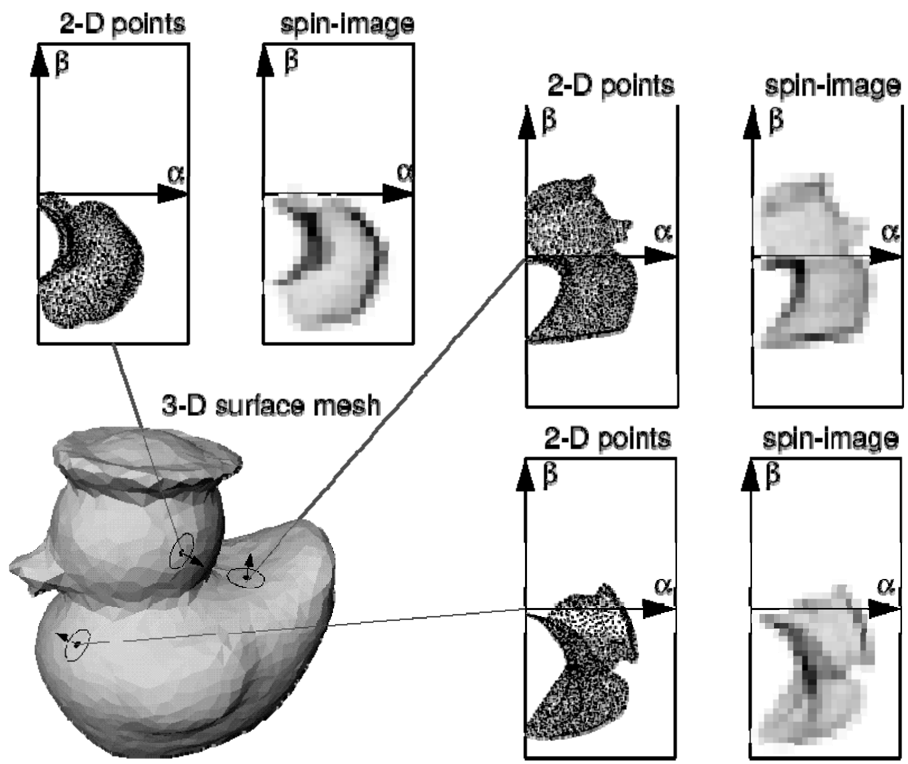
Creates an image associated with a neighborhood of a point.

Compare points by comparing their *spin images* (2D).

Given a point and a normal, every other point is indexed by two parameters:

β distance to tangent plane

α distance to normal line

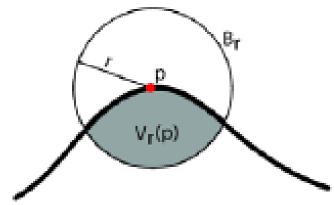


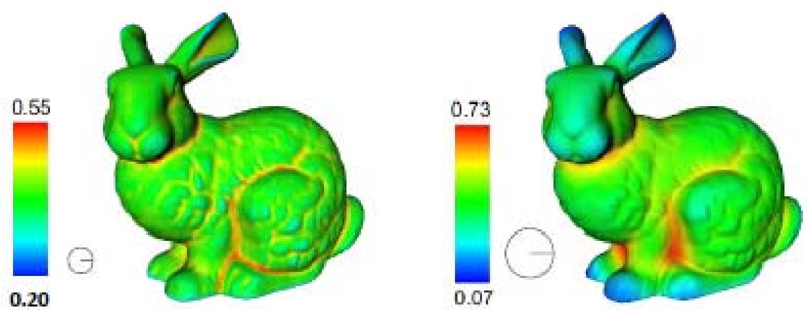
Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes
Johnson et al, PAMI 99

Integral Volume Descriptor

Integral invariant signatures, Manay et al. ECCV 2004
Integral Invariants for Robust Geometry Processing, Pottmann et al. 2007-2009

$$V_r(p) = \int_{B_r(p)\cap S} dx$$





Relation to mean curvature

$$V_r(\mathbf{p}) = \frac{2\pi}{3}r^3 - \frac{\pi H}{4}r^4 + O(r^5)$$

Robust Global Registration,
Gelfand et al. 2005

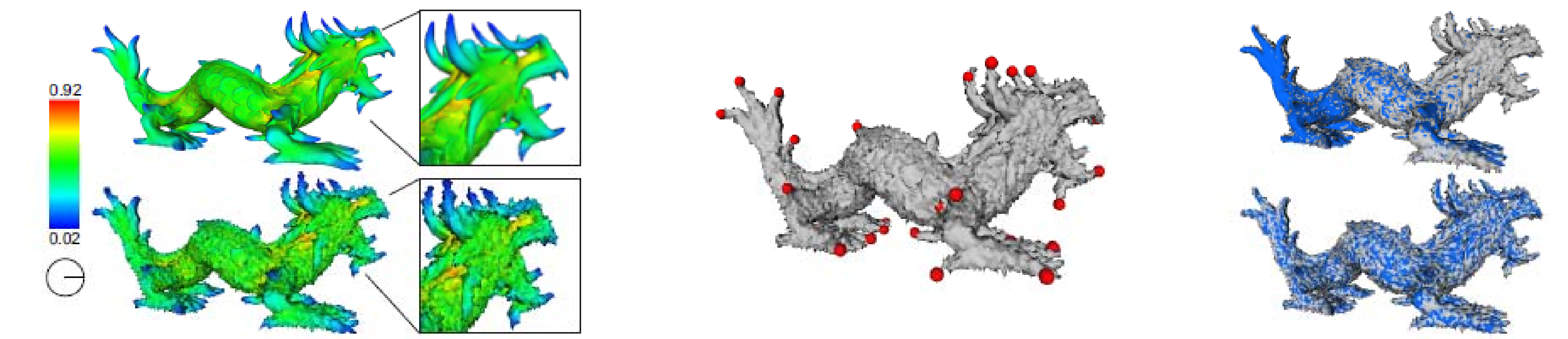
Feature Based Methods

Once we have a feature descriptor, we can find the most *unusual* one: feature detection.

Establish correspondences by first finding *reliable* ones.

Propagate the matches everywhere.

To backtrack use branch-and bound.



Robust Global Registration,
Gelfand et al. 2005

Method Taxonomy

Local vs. Global

refinement (e.g. ICP) | alignment (search)

Rigid vs. Deformable

rotation, translation | general deformation

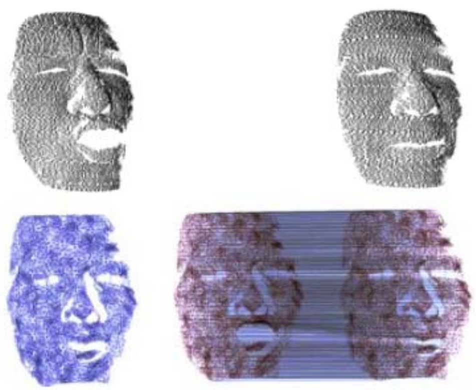
Pair vs. Collection

two shapes | multiple shapes

Deformable Shape Matching

No shape is completely deformable. Every deformable shape matching method uses *some* deformation model.

- Local Deformable Matching:



Wand et al. *SGP '07*

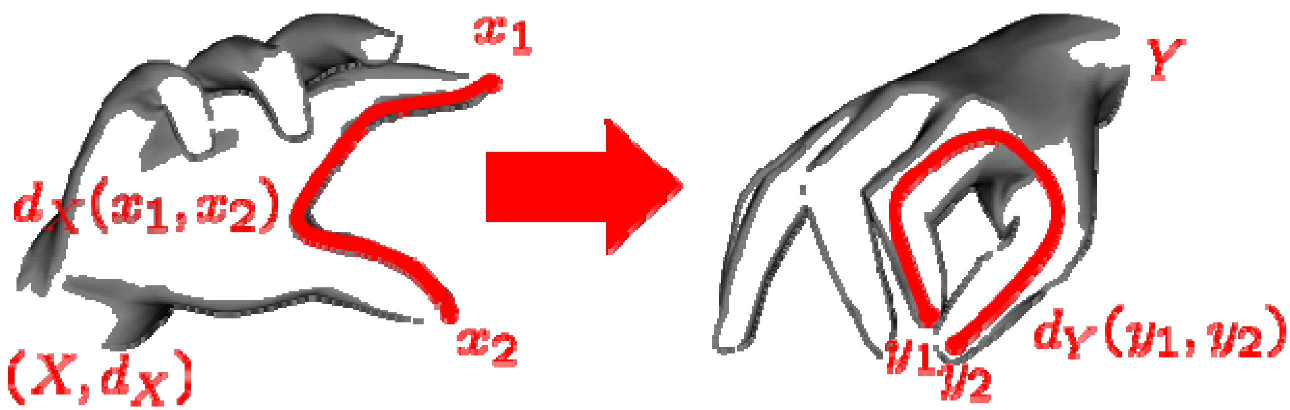
Li et al. *SGP '08* ...

Each coordinate becomes an unknown. Use *smoothness*, elasticity constraints. Resulting problem is always *non-linear*.

Deformable Shape Matching

- Global/Local Deformable Matching Models:

  Most common deformation model: **intrinsic isometries.**
  A deformation that preserves *geodesic* distances.



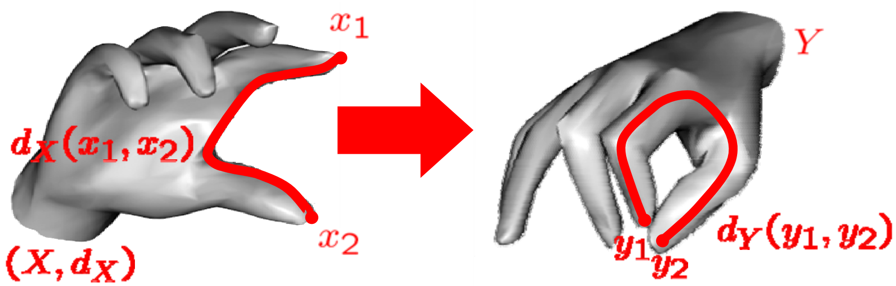
  Recall that rigid deformations preserve *extrinsic* distances.

  Isometric deformation model is useful in matching humans, animals, cloth, some molecular data, etc.

Related Work

- Most global deformable matching methods use isometric deformation model.

GMDS



$$\min_{\{y_1,\ldots y_n\} \subset Y} \|d_X(x_i, x_j) - d_Y(y_i, y_j)\|$$

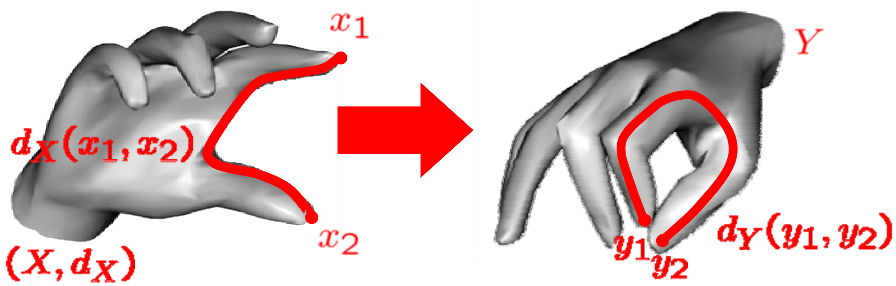
Bronstein et al. PNAS '06

Optimization is non-convex, performed on a *small* subset of points. Need an initial guess.

Related Work

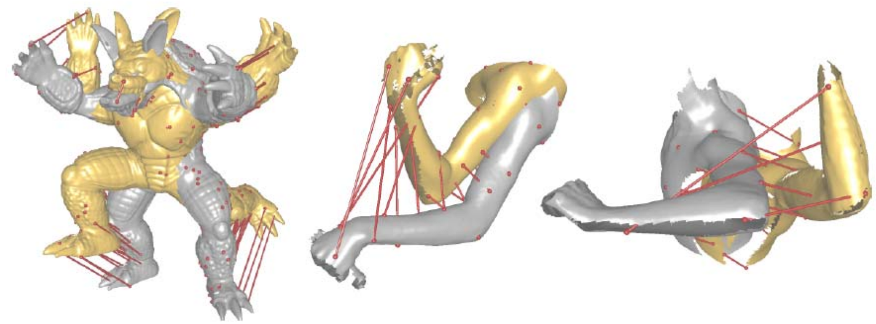
- Most global deformable matching methods use isometric deformation model.

GMDS



$$\min_{\{y_1,\ldots y_n\}\subset Y} \|d_X(x_i, x_j) - d_Y(y_i, y_j)\|$$

Bronstein et al. PNAS ‘06



Huang et al. *SGP ‘08*

Find a few landmark correspondences, extend them using geodesic coordinates.

Deformable Shape Matching

- To define descriptors identify what is preserved under the given class of deformations.

  Isometric deformations preserve geodesic distances. And:

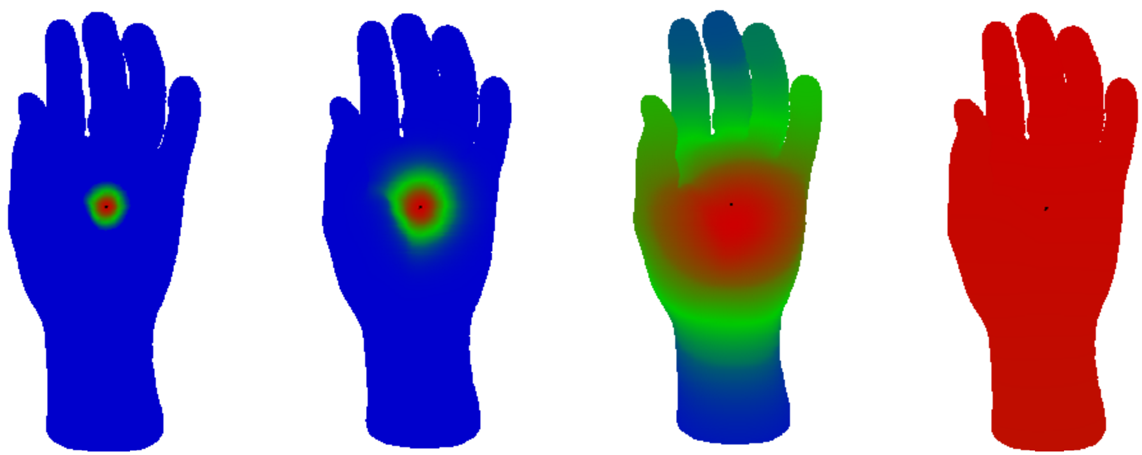
  1. Gaussian Curvature at each point.
  2. Laplace Beltrami Operator and its properties (eigenvalues, eigenvectors).
  3. Heat Diffusion process.

Heat Equation on a Manifold

Heat kernel $k_t(x, y) : \mathbb{R}^+ \times \mathcal{M} \times \mathcal{M} \to \mathbb{R}$

$$f(x,t) = \int_{\mathcal{M}} k_t(x,y) f(y,0)\, dy$$

$k_t(x, y)$: amount of heat transferred from x to y in time t.

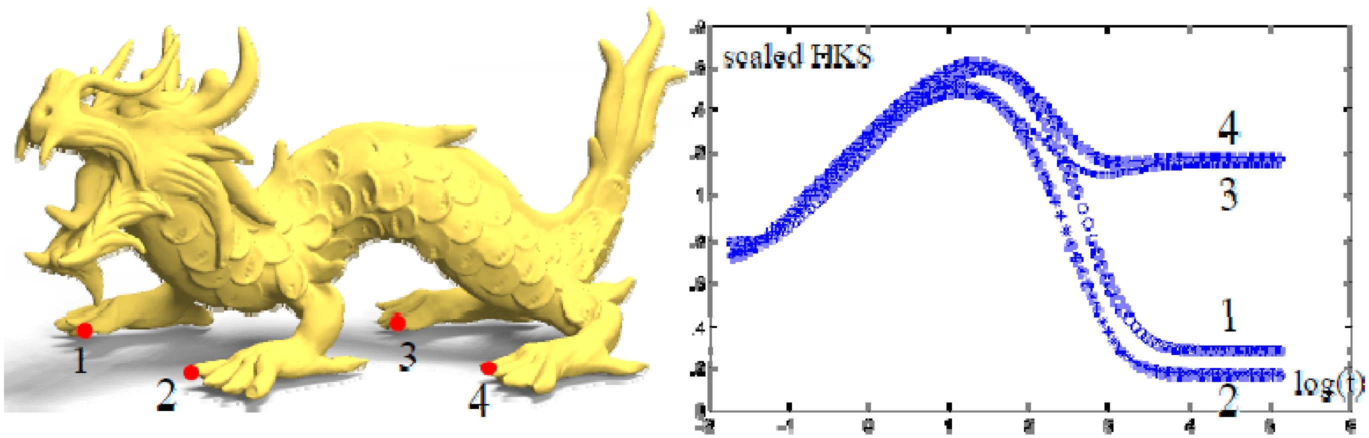


$t = 0.001$ $t = 0.02$ $t = 3$ $t = 40$

Heat Kernel Signature

HKS(p) = $k_t^M(p,p) : \mathbb{R}^+ \to \mathbb{R}^+$: amount of heat left at p at time t.

Signature of a point is a function of one variable.



Invariant to isometric deformations. Moreover complete:

Any continuous map between shapes that preserves HKS must preserve *all* distances.

A Concise and Provably Informative …, Sun et al., SGP 2009

Conclusion

- Shape Matching is an active area of research.
- Local rigid matching works well. Many approaches to global matching. Work well depending on the domain.
- Non-rigid matching is much harder. Isometric deformation model is common, and useful but limiting.
- Research problems: other deformation models, consistent matching with many shapes, robust deformable matching.