Geometric Registration for Deformable Shapes

3.3 Advanced Global Matching

Correlated Correspondences [ASP*04]

A Complete Registration System [HAW*08]

31st Annual Conference of the
European Association for Computer Graphics
eurographics 2010

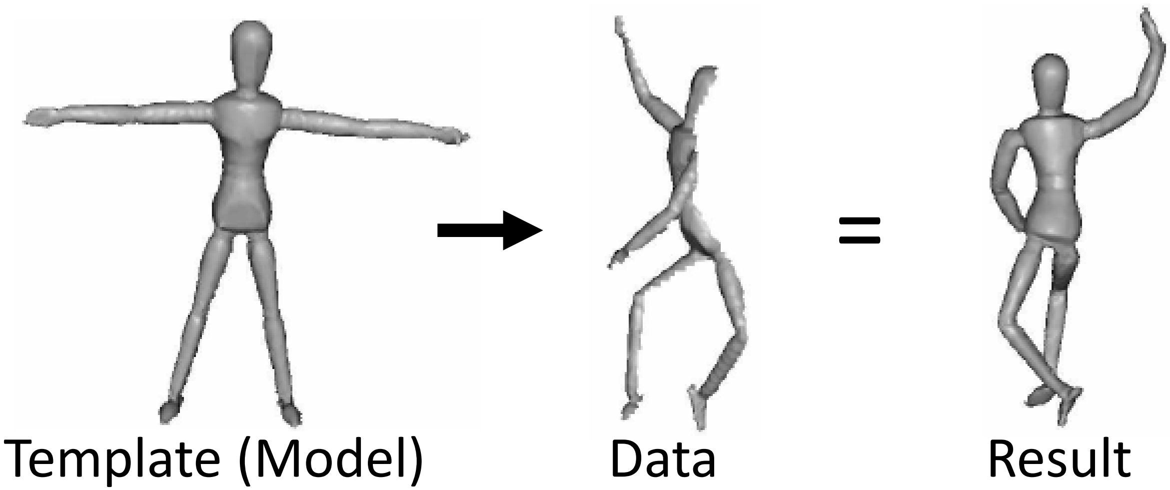
In this session

Advanced Global Matching

- Some practical applications of the optimization presented in the last session
- Correlated Correspondences [ASP*04]: Applies MRF model
- A Complete Registration System [HAW*08]: Applies Spectral matching to filter correspondences

Correlated correspondences

- Correspondence between data and model meshes
- Model mesh is a template; i.e. data is a subset of model



- Not a registration method; just computes corresponding points between data/model meshes
  - Non-rigid ICP [Hanhel et al. 2003] (using the outputted correspondences) used to actually generate the registration results seen in the paper

Basic approach

A joint probability model represents preferred correspondences

- Define a "probability" of each correspondence set between data/model meshes
- Find the correspondence with the highest probability using Loopy Belief Propagation (LBP) [Yedidia et al. 2003]

2 main components (next parts of the talk)

- Probability model
- Optimization

Joint Probability Model

$$P(\{c\}) = \frac{1}{Z} \prod_{k,l} \psi_{kl}(c_k, c_l) \prod_{k} \psi_k(c_k)$$

Compatibility constraints

- Involves pair of correspondences
- Represents prior knowledge of which correspondence sets makes sense

A. Minimize the amount of deformation induced by the correspondences

B. Preserve the geodesic distances in model and data

Singleton constraints

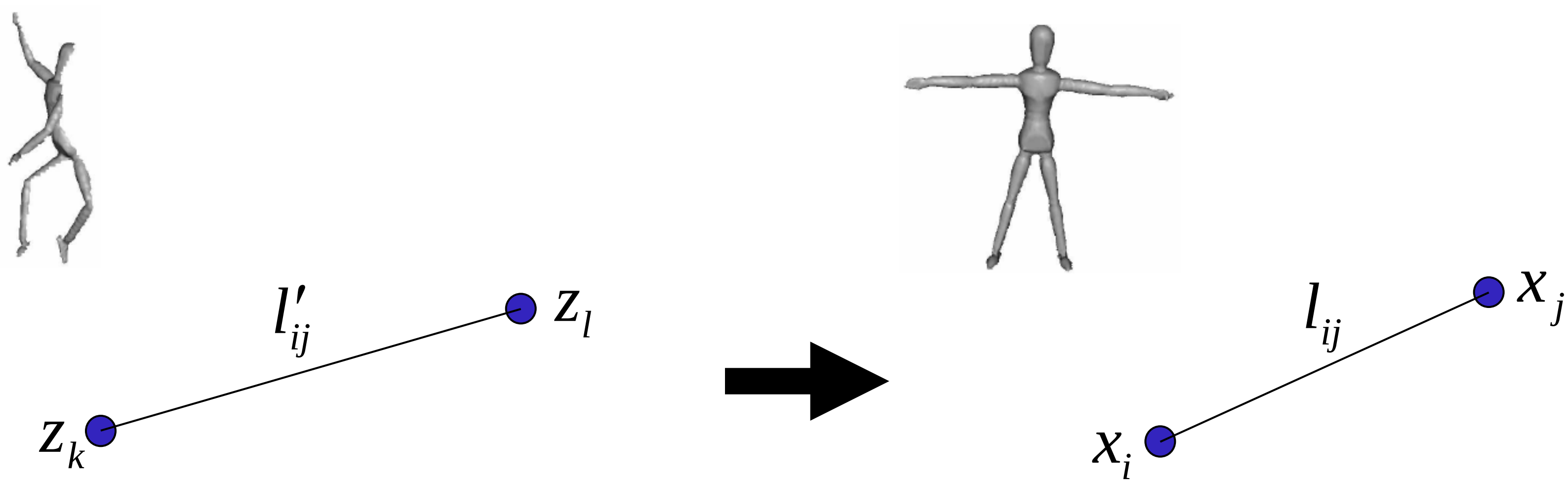
- Involves a single correspondence

C. Corresponding points have same feature descriptor values

Compatibility 1: Deformation potential

Penalize unnatural deformations

- Edges lengths should stay the same $l_{ij} \approx l'_{ij}$



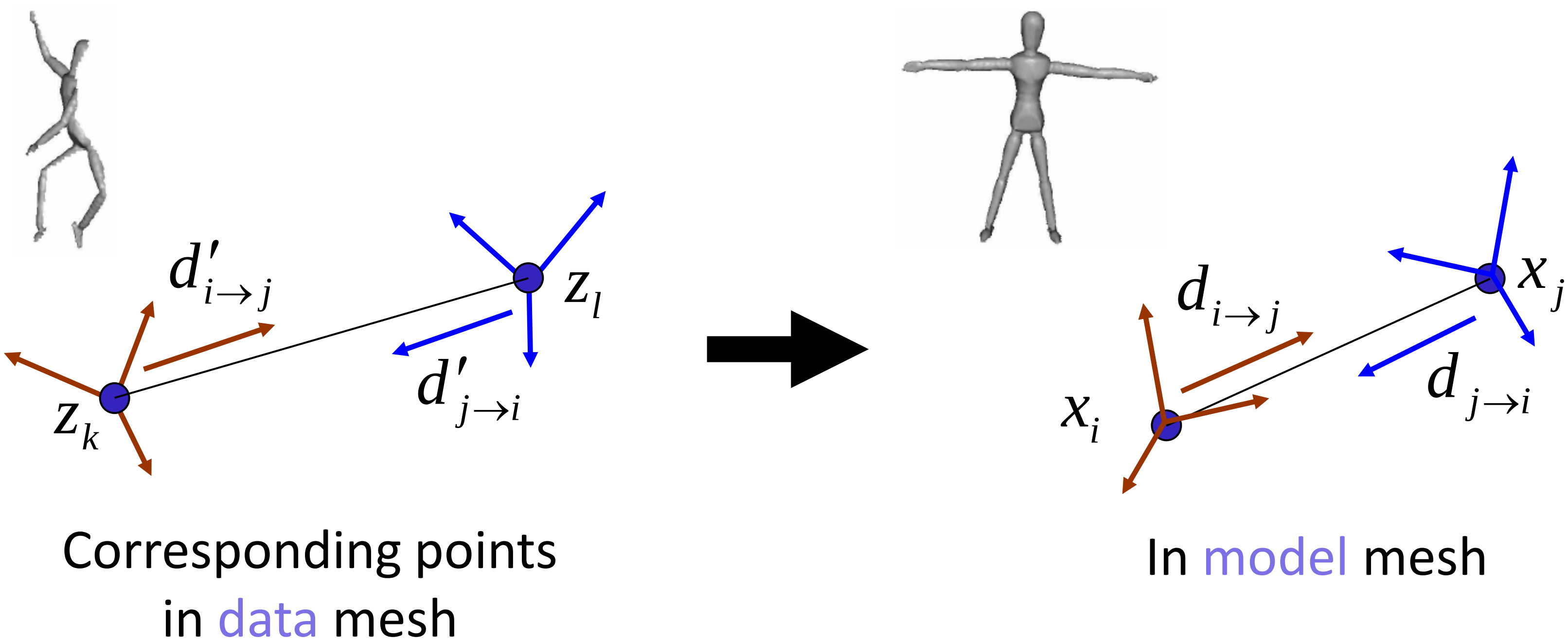
Corresponding points in data mesh

In model mesh

Compatibility 1: Deformation potential

Penalize unnatural deformations

- Edges should twist little as possible $d_{i\to j} \approx d'_{i\to j},\ d_{j\to i} \approx d'_{j\to i}$
- $d_{i\to j}$ Is the direction from x_i to x_j in x_i's coord system



Encoding the preference

- Zero-mean Gaussian noise model for length and twists
- Define potential ψ_d for each edge (z_k, z_l) in the data mesh
  - (c_k, c_l) are "correspondence variables" indicating what is the corresponding point in the model mesh for z_k, z_l respectively

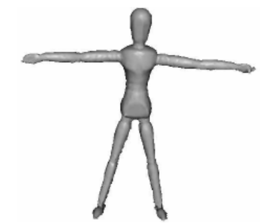
$$\psi_d(c_k = i, c_l = j) = G(l'_{ij} \mid l_{ij}) G(d'_{i \to j} \mid d_{i \to j}) G(d'_{j \to i} \mid d_{j \to i})$$

- Caveat: additional rotation needed to measure twist
  - For each possibility of $c_k = i$ precompute aligning rotation matrices via rigid ICP on surrounding local patch
  - Expand corresp. variables to be site/rotation pairs

Compatibility 2: Geodesic distance potential

Penalize large changes in geodesic distance

- Geodesically nearby points should stay nearby
  - Enforced for each edge in the data mesh



z_l

z_k

$$dist_{Geodesic}(z_k, z_l) \approx p$$

x_j

x_i

$$dist_{Geodesic}(x_i, x_j)$$

If > 3.5p → prob assigned 0
otherwise → prob assigned 1

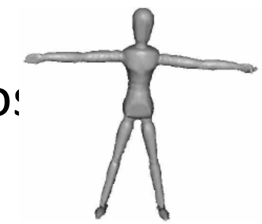
Adjacent points
in data mesh

Corresponding points
in model mesh

Compatibility 2: Geodesic distance potential

Penalize large changes in geodesic distance

- Geodesically far points should stay far away
  - Enforced for each pair of points in the data mesh whos
    geodesic distance is > 5p



z_l z_k

$dist_{Geodesic}(z_k, z_l) > 5p$

x_j x_i

$dist_{Geodesic}(x_i, x_j)$

If < 2p → prob assigned 0
otherwise → prob assigned 1

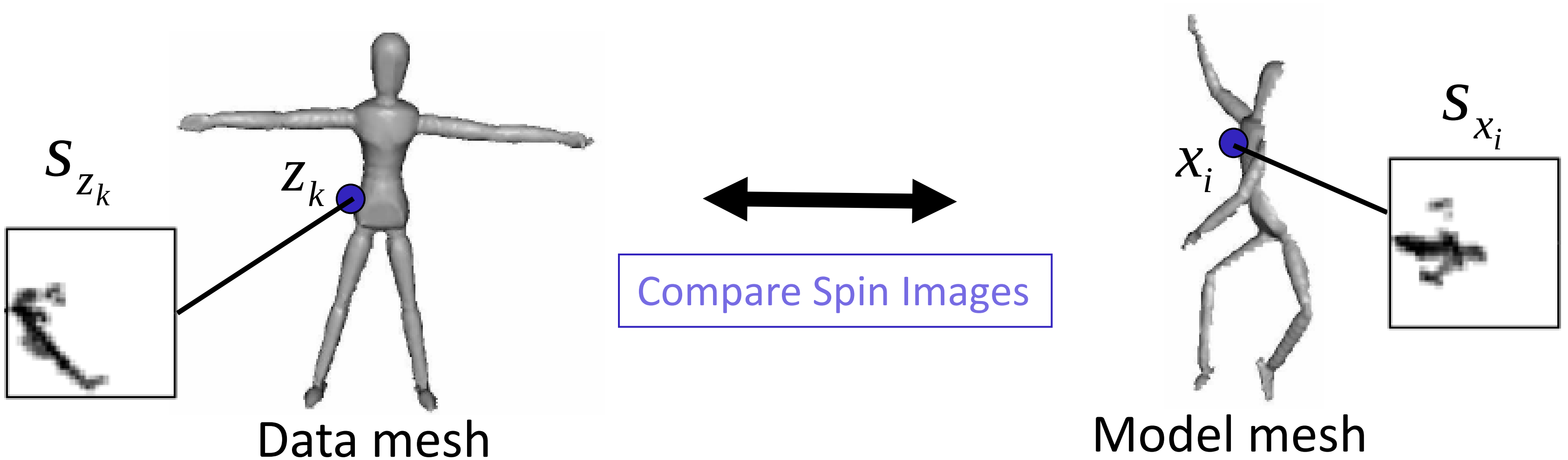
Adjacent points
in data mesh

Corresponding points
in model mesh

Singletons: Local surface signature potential

Spin images gives matching score for each individual correspondence

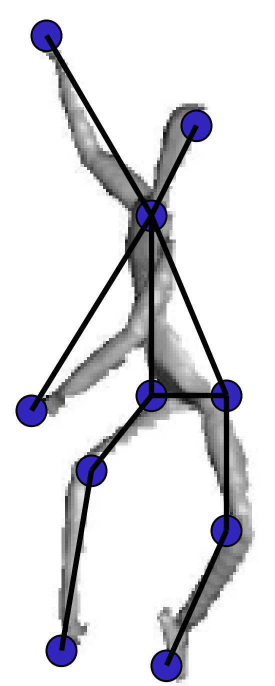
- Compute spin images & compress using PCA
  → gives surface signature s_{x_i} at each point x_i
- Discrepancy between s_{z_k} (data) and s_{x_i} (model)
- Zero-mean Gaussian noise model



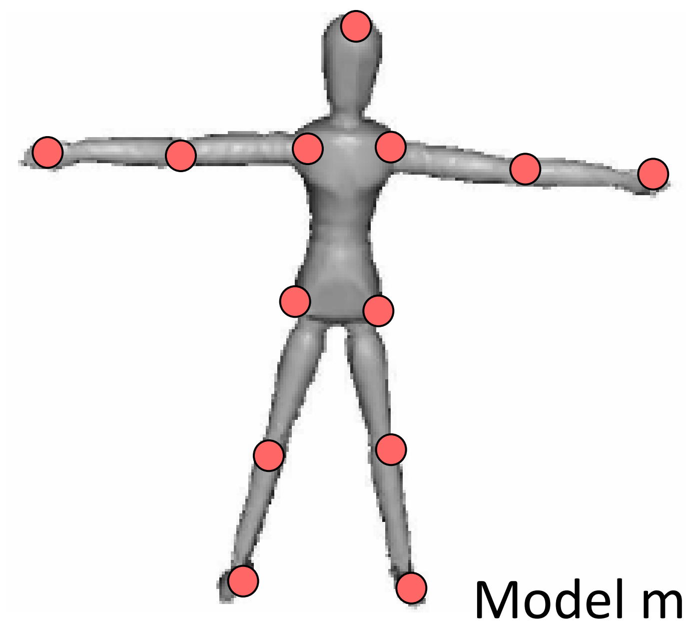
Model summary

Get Pairwise Markov Random Field (MRF)

- Pointwise potential for each pt in data
- Pairwise potential for each edge in data
    - Far geodesic potentials for each pair of points > 5p apart



Data mesh

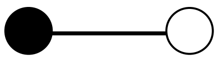


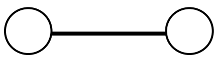
Model mesh

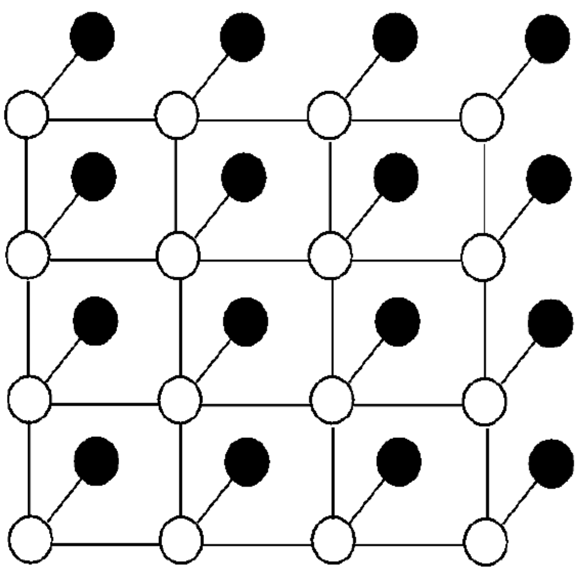
Quick intro: Markov Random Fields

Joint probability function visualized by a graph

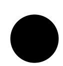
- Prob. = Product of the potentials at all edges

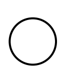
 $\psi_k(c_k)$ ← (ex) Surface signature potential

 $\psi_{kl}(c_k, c_l)$ ← (ex) Deformation, geodesic distance potential



$$P(\{c\}) = \frac{1}{Z} \prod_{k,l} \psi_{kl}(c_k, c_l) \prod_k \psi_k(c_k)$$

 “Observed” nodes

 “Hidden” nodes

Loopy Belief Propagation (LBP)

Compute marginal probability for each variable

- Pick variable value that maximizes the marginal prob.

Usual way to compute marginal probabilities (tabulate and sum up) takes exponential time

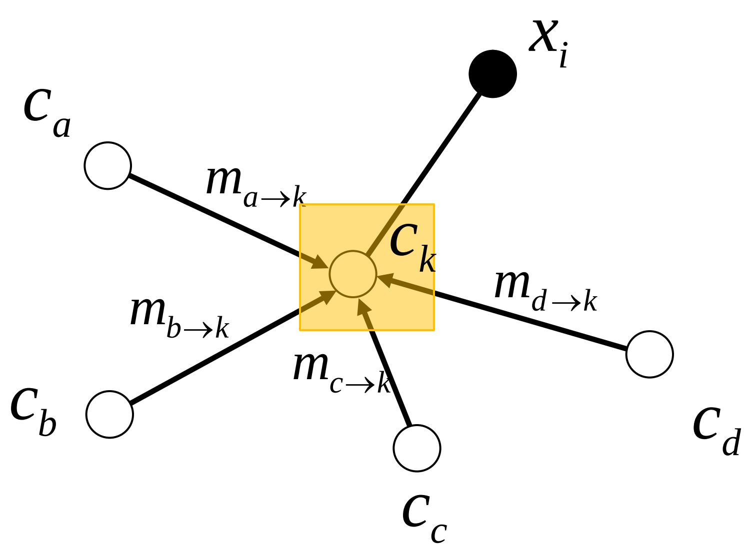
- BP is a dynamic programming approach to efficiently compute marginal probabilities
- Exact for tree MRFs, approximate for general MRFs

Loopy Belief Propagation (LBP)

Basic idea

- Marginals at node proportional to product of pointwise potential and incoming messages

$$b_k(c_k) = k\phi_k(c_k) \prod_{l \in N(k)} m_{l \to k}(c_k)$$

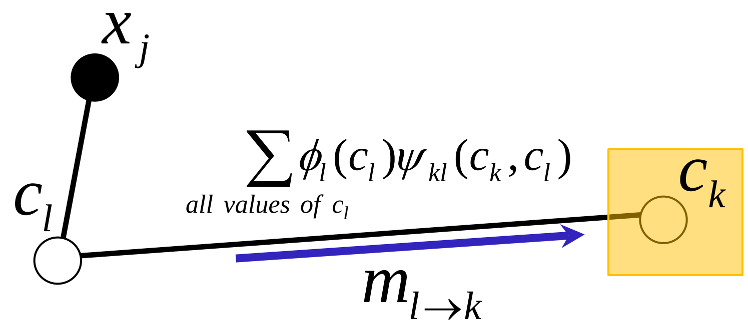


Loopy Belief Propagation (LBP)

Basic idea

- Compute these messages (at each edge) and we are done

$$m_{l \to k}(c_k) \leftarrow \sum_{all\ values\ of\ c_l} \phi_l(c_l)\psi_{kl}(c_k, c_l) \prod_{q \in N(l) \backslash k} m_{q \to l}(c_l)$$

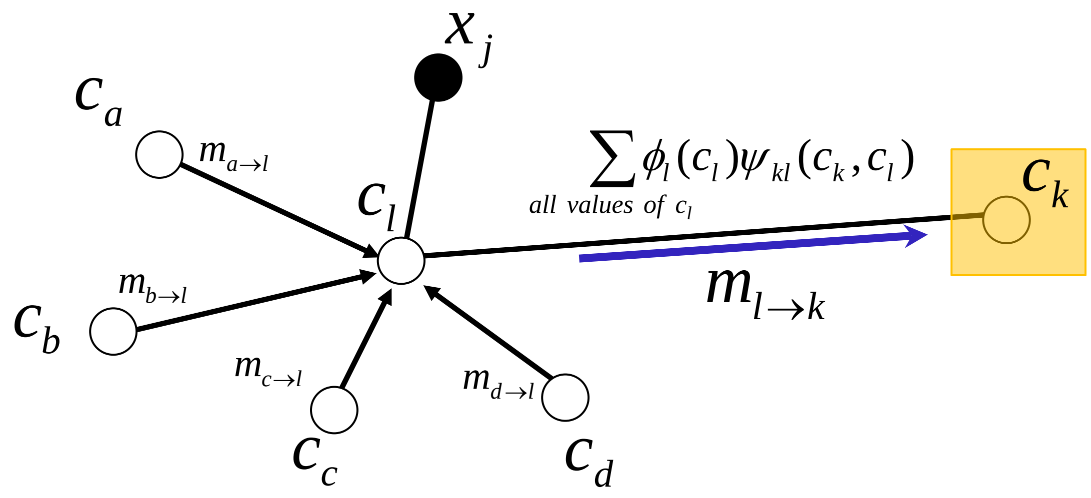


Loopy Belief Propagation (LBP)

Basic idea

- Compute these messages (at each edge) and we are done

$$m_{l\to k}(c_k) \leftarrow \sum_{\text{all values of } c_l} \phi_l(c_l)\psi_{kl}(c_k, c_l) \prod_{q\in N(l)\backslash k} m_{q\to l}(c_l)$$

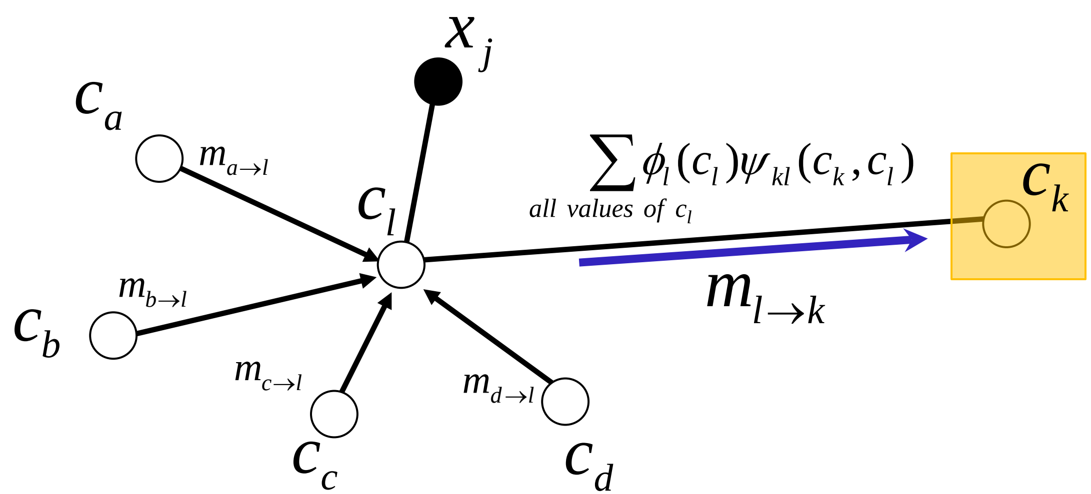


Loopy Belief Propagation (LBP)

Basic idea

- Compute these messages (at each edge) and we are done

$$m_{l\to k}(c_k) \leftarrow \sum_{all\ values\ of\ c_l} \phi_l(c_l)\psi_{kl}(c_k, c_l) \prod_{q\in N(l)\setminus k} m_{q\to l}(c_l)$$

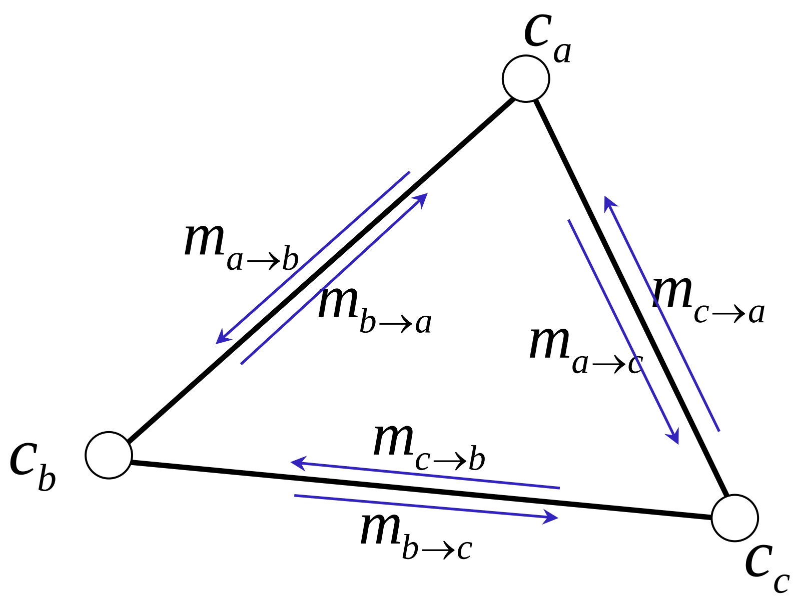


- Recursive formulation
- Start at ends and work your way towards the rest

Loopy Belief Propagation (LBP)

Loops: iterate until messages converge

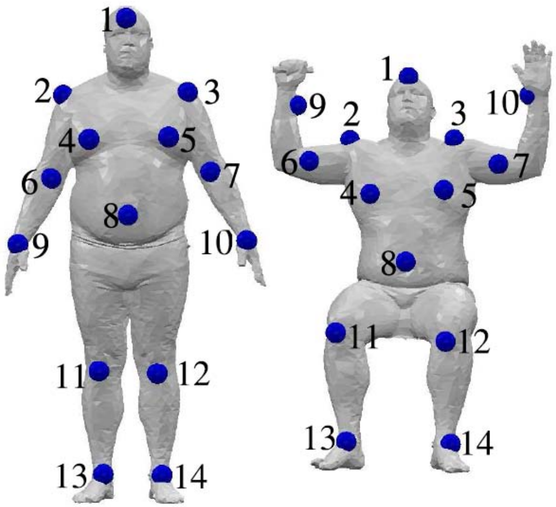
- Start with initial values (ex: $\sum_{all\ values\ of\ c_a} \phi_a(c_a)\psi_{ab}(c_a, c_b)$)
- Apply message update rule until convergence



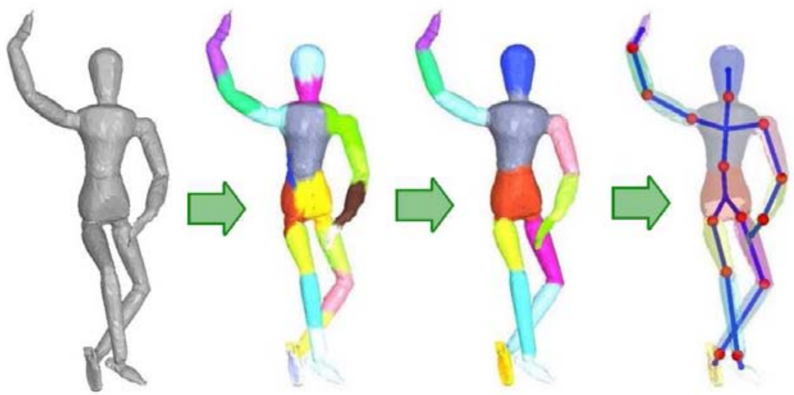
- Convergence not guaranteed, but works well in practice

Results & Applications

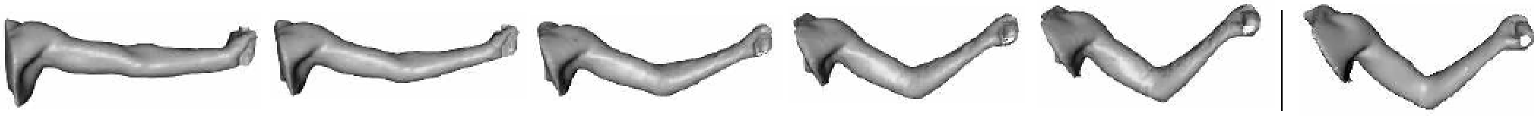
- Efficient, coarse-to-fine implementation
- Xeon 2.4 GHz CPU, 1.5 mins for arm, 10 mins for puppet



Correspondences on human body models



Finding articulated parts



Interpolation between poses

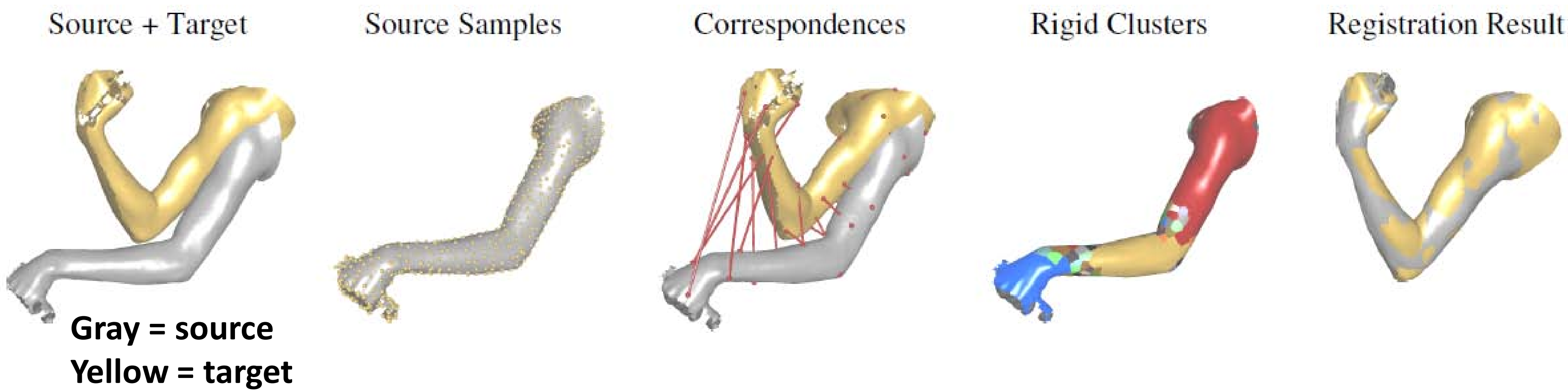
Next topic: HAW*08

An application to the spectral matching method of last session

- A good illustration of how a matching method fits into a real registration pipeline

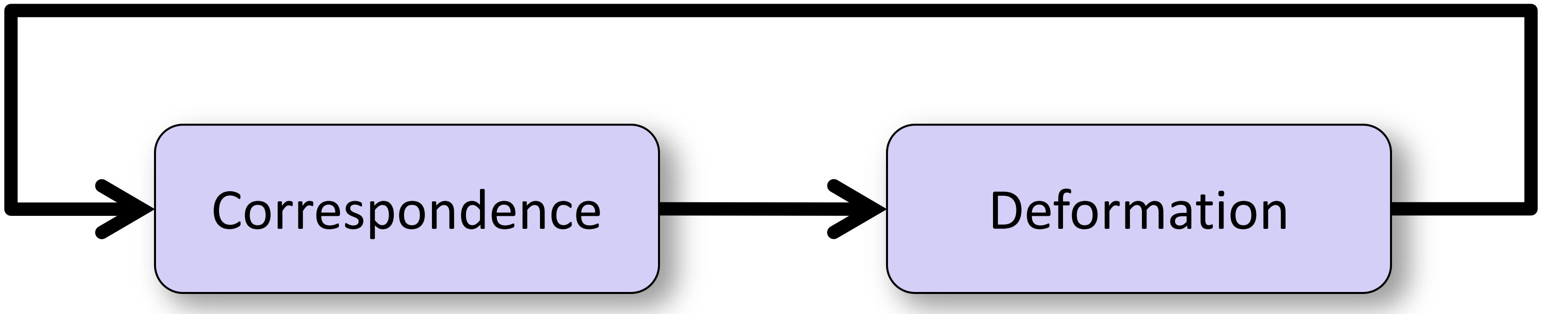
A pairwise method

- Deform the source shape to match the target shape



Overview

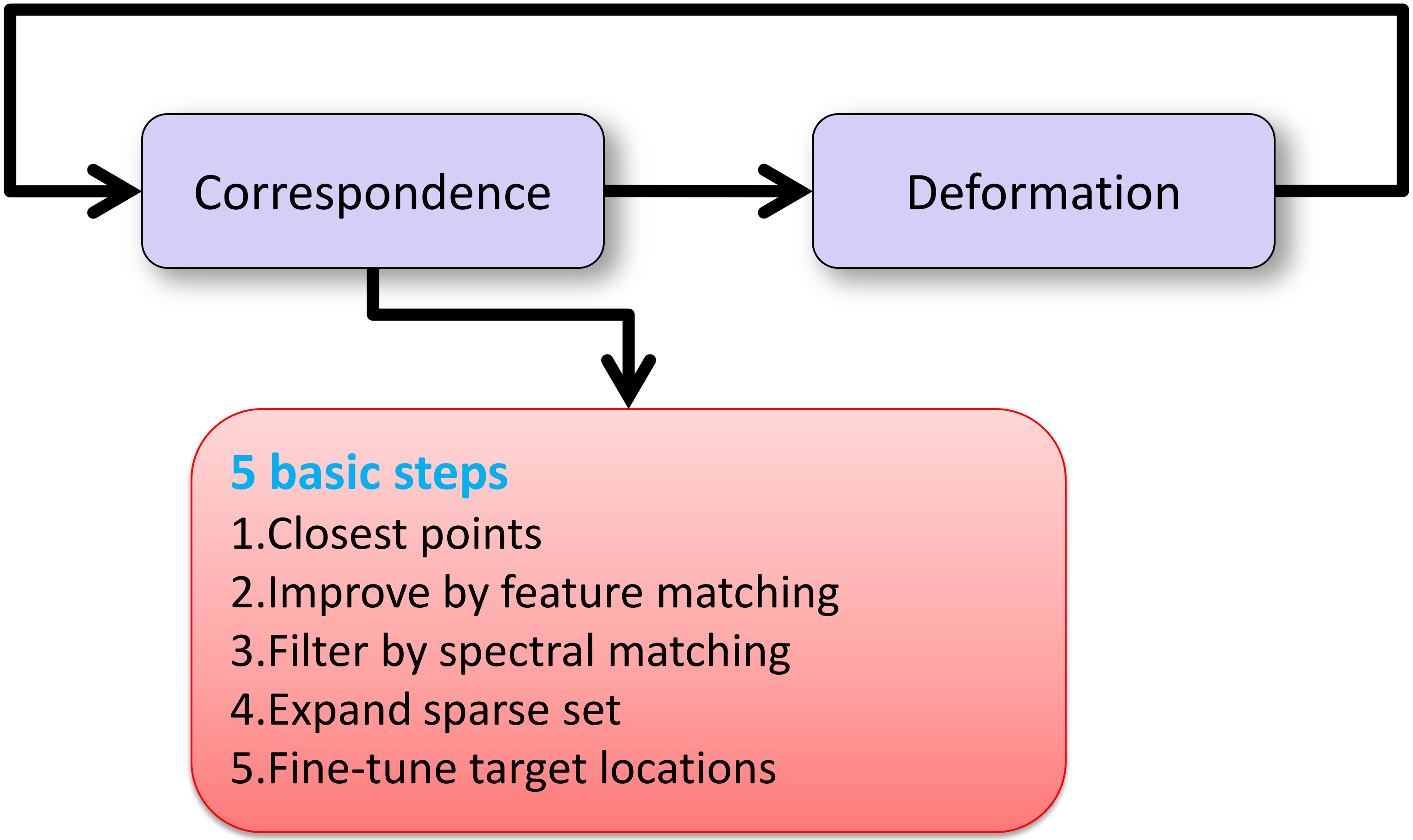
Performs both correspondence and deformation



- Correspondences based on improving closest points
- After finding correspondences, deform to move shapes closer together
- Re-take correspondences from the deformed position
- Deform again, and repeat until convergence

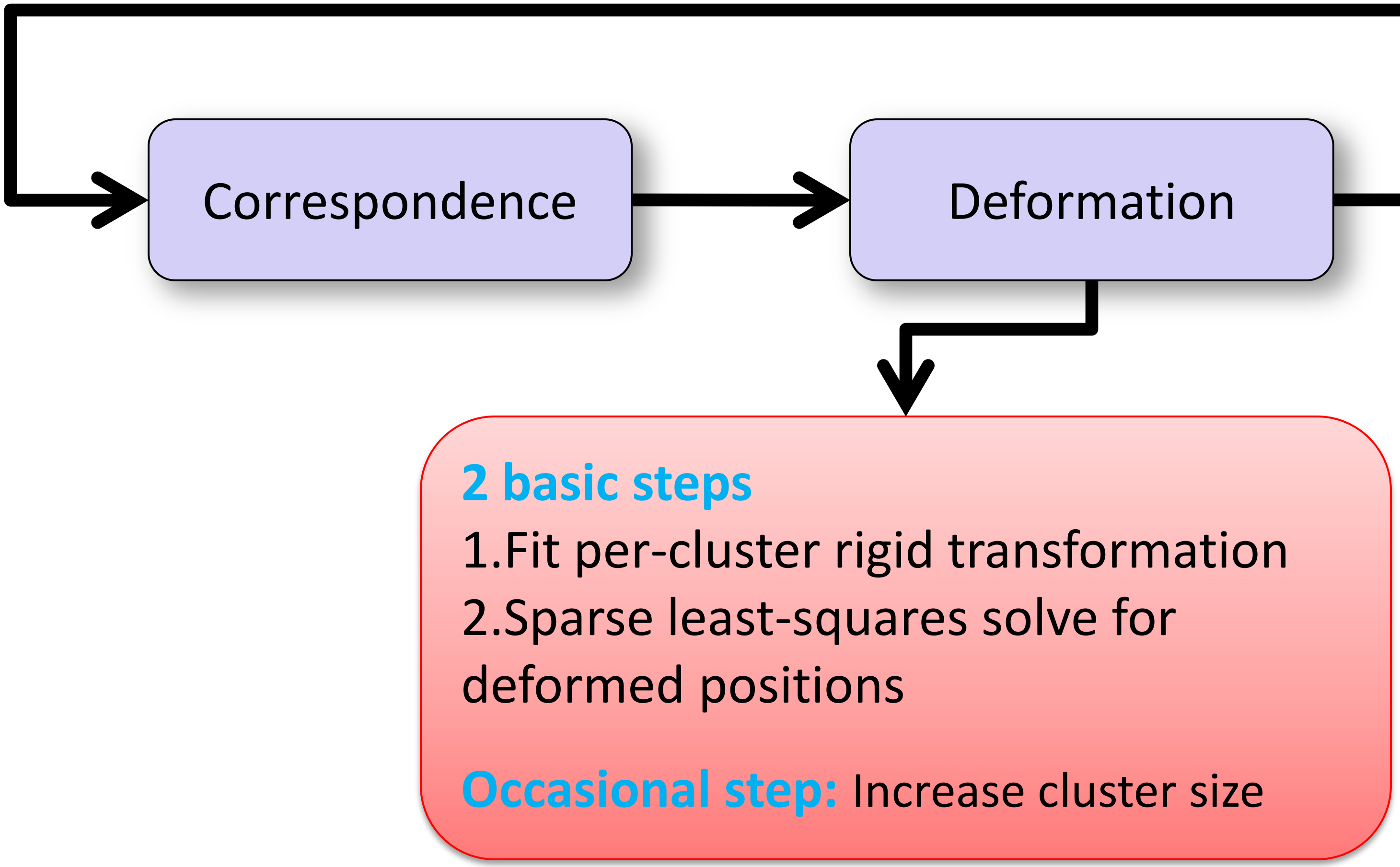
Overview

Performs both correspondence and deformation



Overview

Performs both correspondence and deformation



Detailed Overview

Sampling

- Whole process works with reduced sample set

Correspondence & Deformation

- Examine each step in more detail

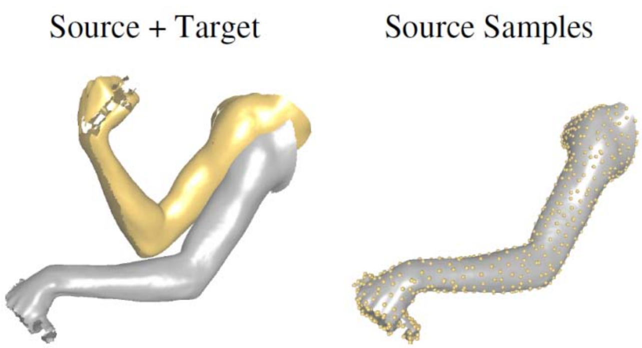
Discussion

- Discuss pros/cons

Sample for robustness & efficiency

Coarse to fine approach

- Use uniform subsampling of the surface and its normals
- Improve efficiency, can improve robustness to local minima



Let's make it more concrete

- Sample set denoted s_i
- In correspondence: for each s_i, find corresponding target points t_i
- In deformation: given t_i, find deformed sample positions s_i' that match t_i while preserving local shape detail

Correspondence Step #1

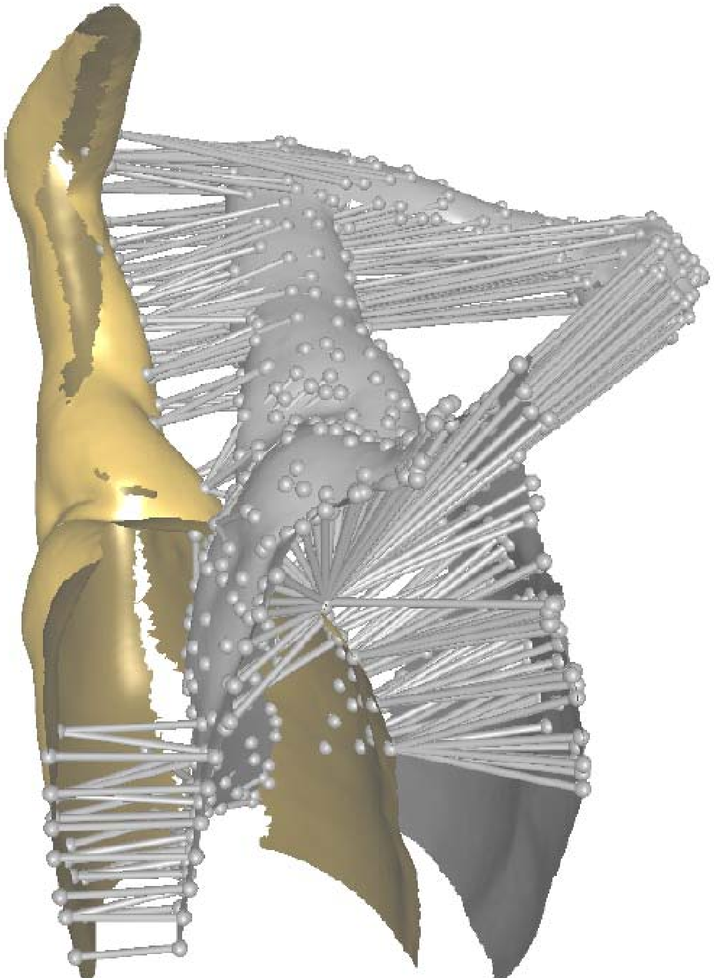
Find closest points

- For each source sample, find the closest target sample
  - s = sample point on source
  - t = sample point on target

$$\underset{t \in \hat{T}}{\arg\min} \left\| s - t \right\|^2$$

- Usually pretty bad

Target (yellow) ← Source (gray)



Closest point correspondences

Correspondence Step #2

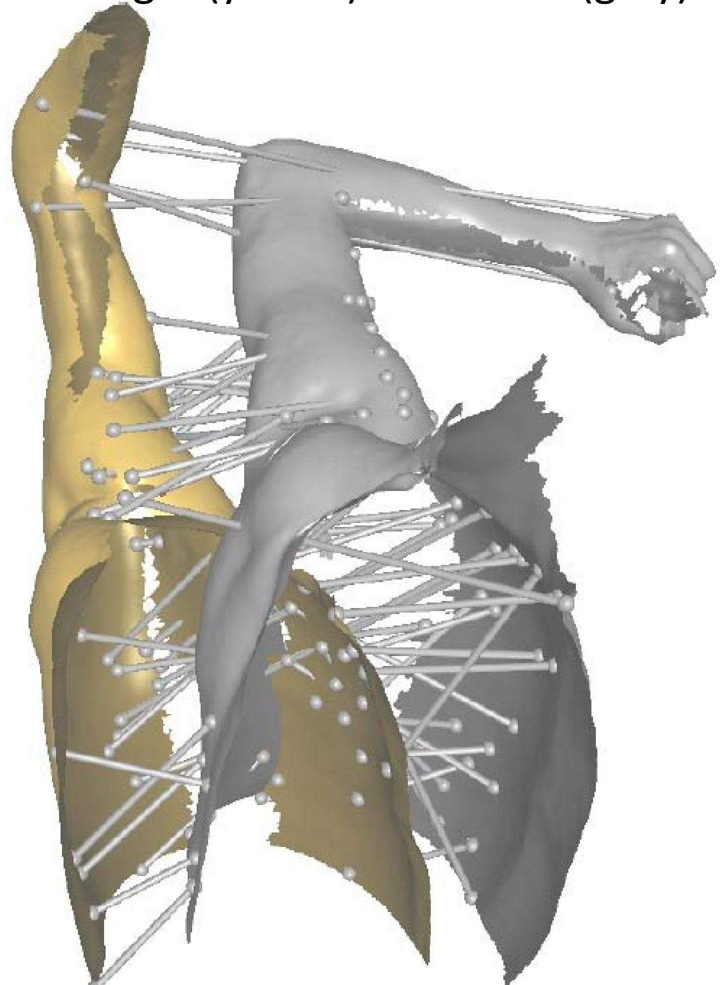
Improve by feature matching

- Search target's neighbors to see if there's better feature match, replace target
  - Let f(s) be feature value of s

$$t \leftarrow \underset{t' \in N(t)}{\arg\min} \left\| f(s) - f(t') \right\|^2$$

- Iterate until we stop moving
- If we move too much, discard correspondence
- Much better, but still outliers

Target (yellow) 🡰 Source (gray)



Feature-matched correspondences

Correspondence Step #3

Filter by spectral matching

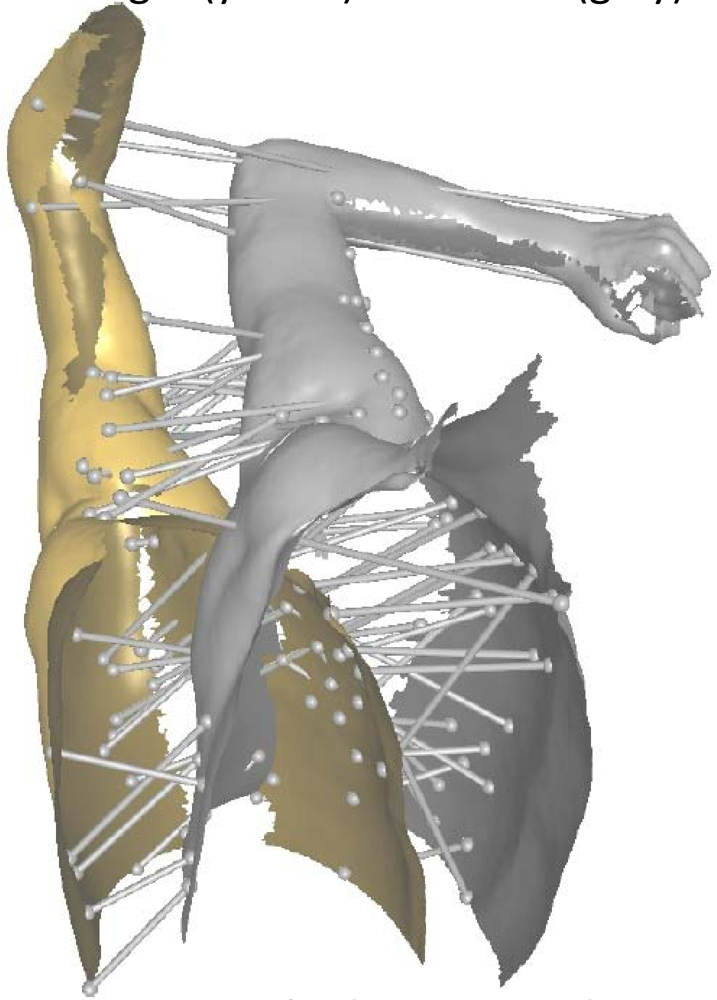
- ***(First some preprocessing)***
- Construct *k*-nn graph on both src & tgt sample set (k = 15)
- Length of shortest path on graph gives approx. geodesic distances on src & tgt

  $$d_g(s_i, s_j) \quad d_g(t_i, t_j)$$

- Goal is to filter these ---------➔

  and keep a subset which is geodesically consistent

Target (yellow) 🡐 Source (gray)



Feature-matched correspondences

Correspondence Step #3

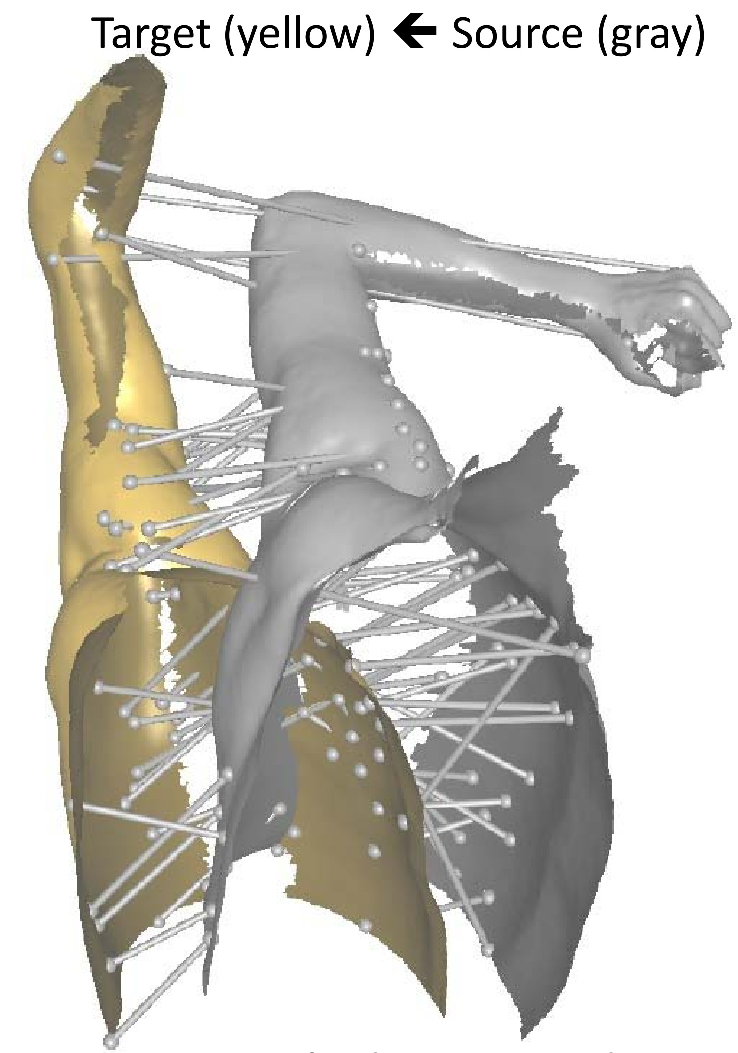
Filter by spectral matching

- Construct affinity matrix M using these shortest path distances
- Consistency term & matrix

$$c_{ij} = \min\{\frac{d_g(s_i, s_j)}{d_g(t_i, t_j)}, \frac{d_g(t_i, t_j)}{d_g(s_i, s_j)}\}, \ c_{ii} = 1$$

$$\mathbf{M}_{ij} = \begin{cases} (\frac{c_{ij} - c_0}{1 - c_0})^2 & c_{ij} > c_0, \\ 0 & \text{otherwise,} \end{cases}$$

  - Threshold c_0 = 0.7 gives how much error in consistency we are willing to accept



Feature-matched correspondences

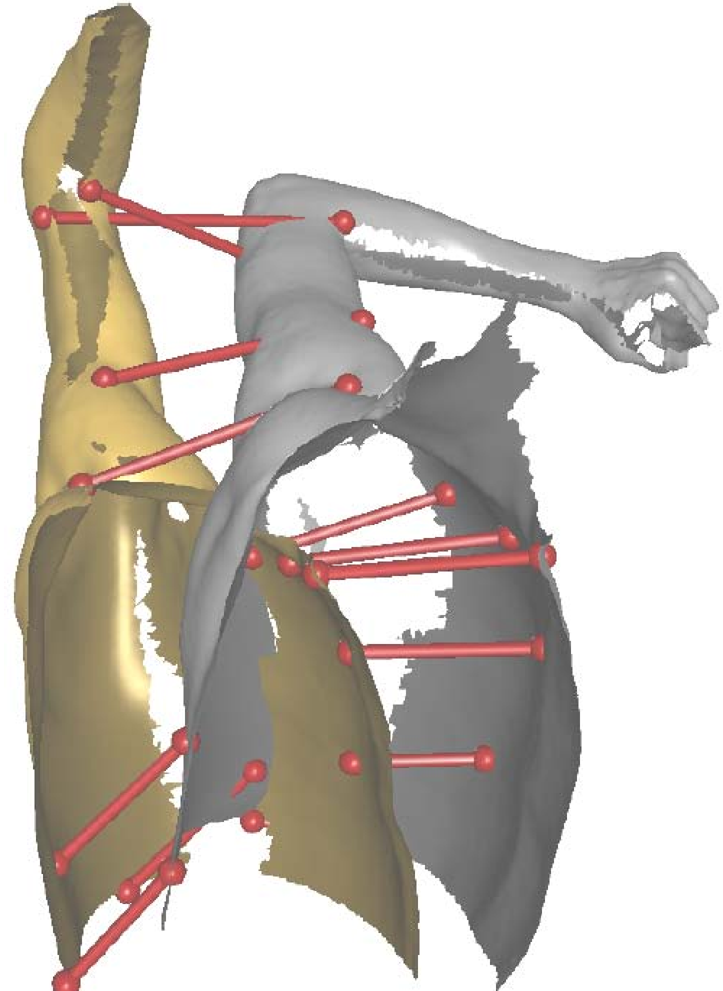
Correspondence Step #3

Filter by spectral matching

- Apply spectral matching: find eigenvector with largest eigenvalue → score for each correspondence
- Iteratively add corresp. with largest score while consistency with the rest is above c_0
- Gives kernel correspondences

- Filtered matches usually sparse

Target (yellow) ← Source (gray)



Filtered correspondences

Correspondence Step #4

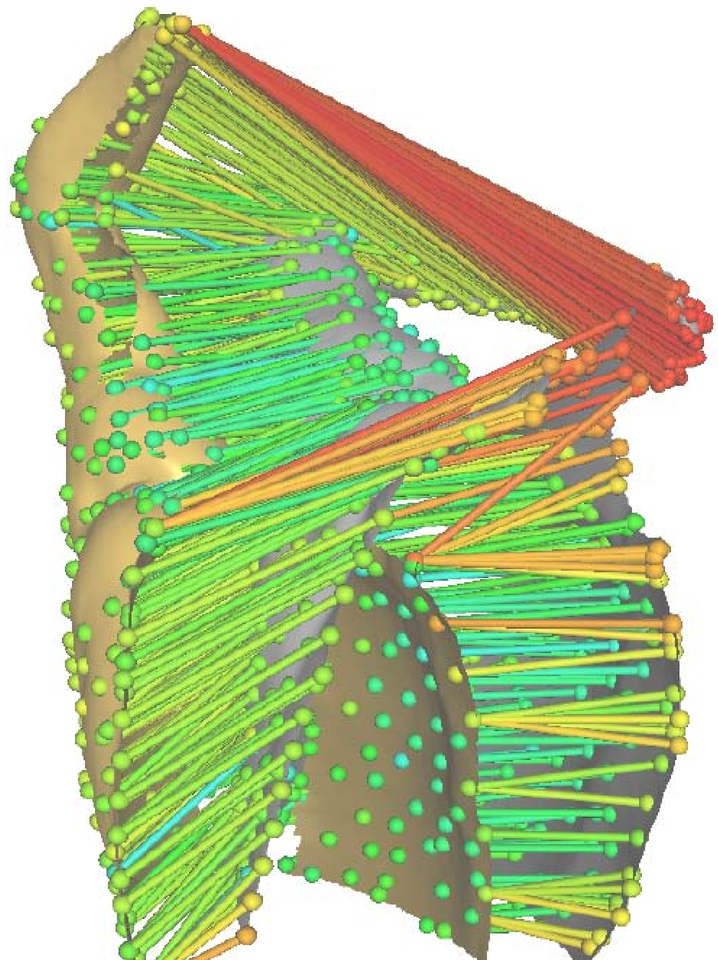
Expand sparse set

- Lots of samples have no target position
- For these, find best target position that respects geodesic distances to kernel set

$$\mathbf{t}_i = \arg\min_{\mathbf{t} \in N_g(\mathbf{t}_j, \bar{T})} e_K(\mathbf{s}_i, \mathbf{t})$$

$$e_K(\mathbf{s}, \mathbf{t}) = \sum_{(\mathbf{s}_k, \mathbf{t}_k) \in K} \left[d_g(\mathbf{s}, \mathbf{s}_k) - d_g(\mathbf{t}, \mathbf{t}_k) \right]^2$$

Target (yellow) 🡠 Source (gray)



Expanded correspondences

Correspondence Step #4

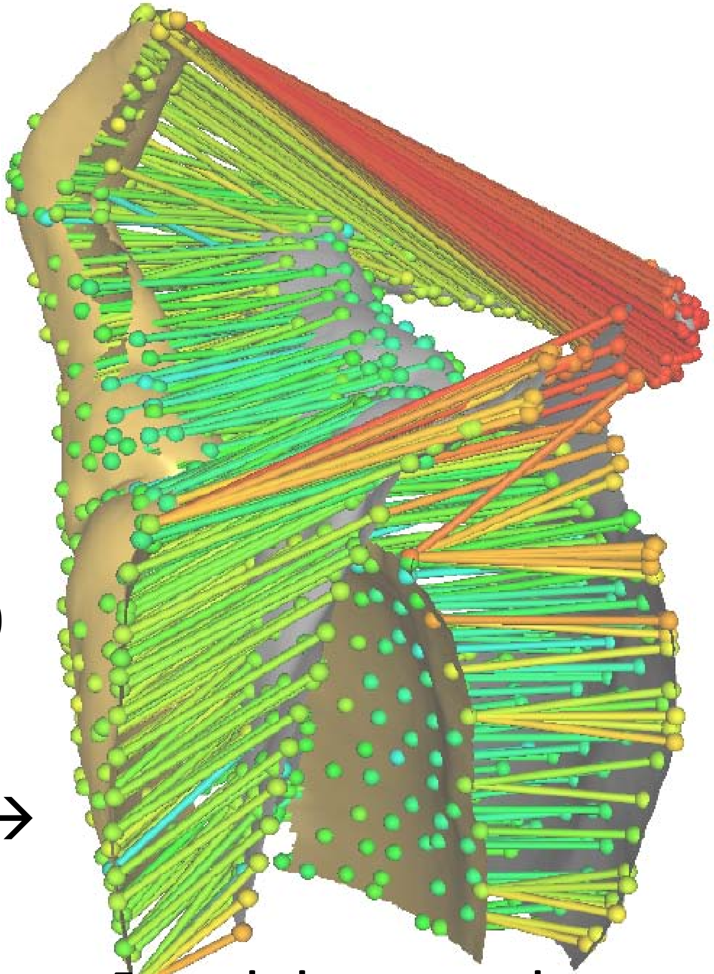
Expand sparse set

- Lots of samples have no target position
- Compute confidence weight based only how well it respects geodesic distances to kernel set

$$w_i = \exp\left(-\frac{e_K(\mathbf{s}_i, \mathbf{t}_i)}{2e}\right) \qquad e = \frac{1}{|K|} \sum_{(\mathbf{s}_k, \mathbf{t}_k) \in K} e_K(\mathbf{s}_k, \mathbf{t}_k)$$

Red = not consistent ---→
Blue = very consistent

Target (yellow) 🡐 Source (gray)



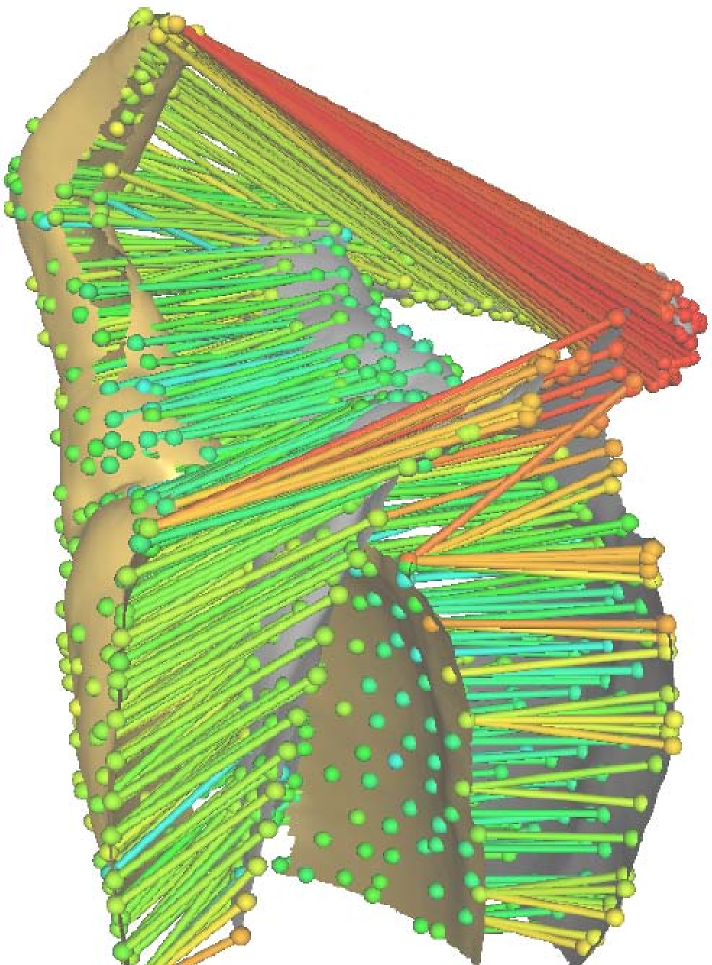
Expanded correspondences

Correspondence Step #5

Fine-tuning

- So far, target points restricted to be points in target samples
- Not accurate when shapes are close together
- Relax this restriction and let target points become any point in the original point cloud
- Replace target sample with a closer neighbor in the original point cloud

Target (yellow) 🡐 Source (gray)



Expanded correspondences

Deformation

Solved by energy minimization (least squares)

- Last step gave target positions t_i
- Now find deformed sample positions s_i' that match target positions t_i

Two basic criteria:

- Match correspondences: s_i should be close to t_i
- Shape should preserve detail (as-rigid-as-possible)
- Combine to give energy term:

$$E = \lambda_{corr} E_{corr} + \lambda_{rigid} E_{rigid}$$

Correspondence matching term

Combination of point-to-point (α=0.6) and point-to-plane (β=0.4) metrics

- Weighted by confidence weight w_i of the target position

$$E_{corr} = \sum_{\mathbf{s}_i \in \bar{S}} w_i \left[\alpha \left\| \mathbf{s}_i' - \mathbf{t}_i \right\|^2 + \beta ((\mathbf{s}_i' - \mathbf{t}_i)^T \mathbf{n}_i)^2 \right]$$

Point-to-point

Point-to-plane

Shape preservation term

Deformed positions should preserve shape detail

- Form an extended cluster $\tilde{C}_k$ for each sample point: the sample itself and its neighbors
- For each $\tilde{C}_k$ find the rigid transformation (R,T) from sample positions to their deformed locations

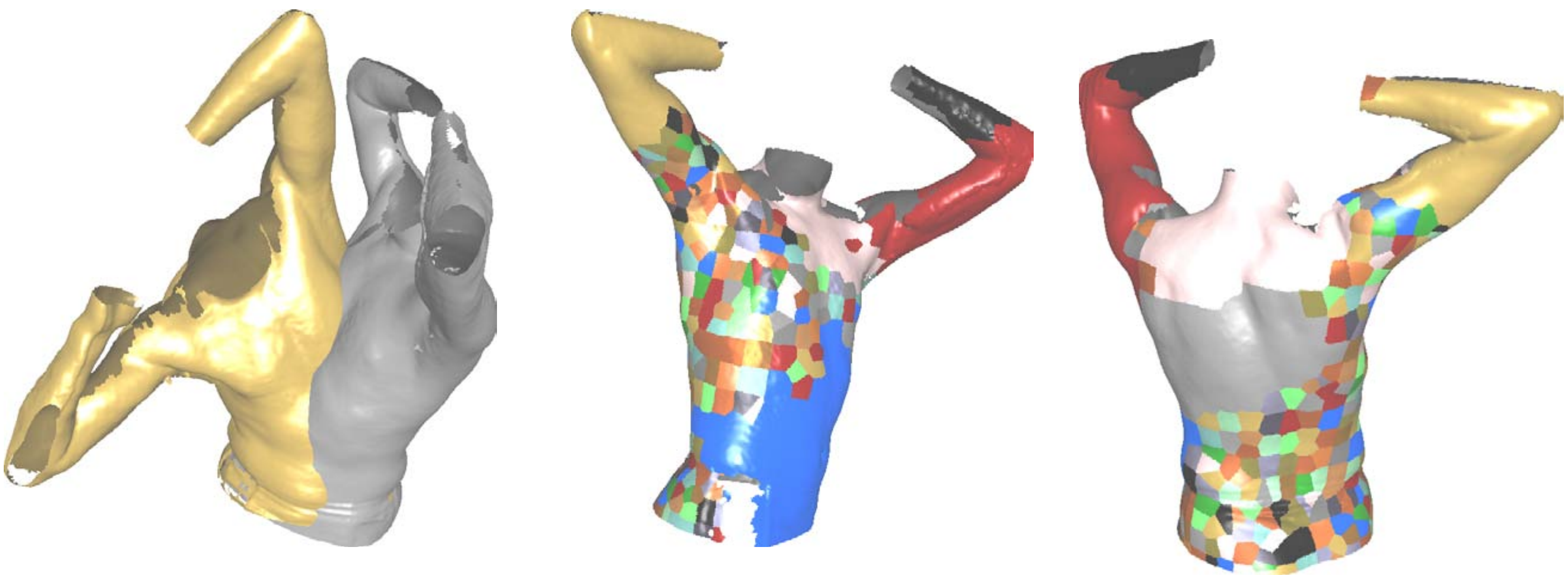
$$E_k = \sum_{s_i \in C_k} \left\| \mathbf{R}_k \mathbf{s}_i + \mathbf{T}_k - \mathbf{s}_i' \right\|^2$$

- When solving for s_i', constrain them to move rigidly according to each cluster that it's associated with

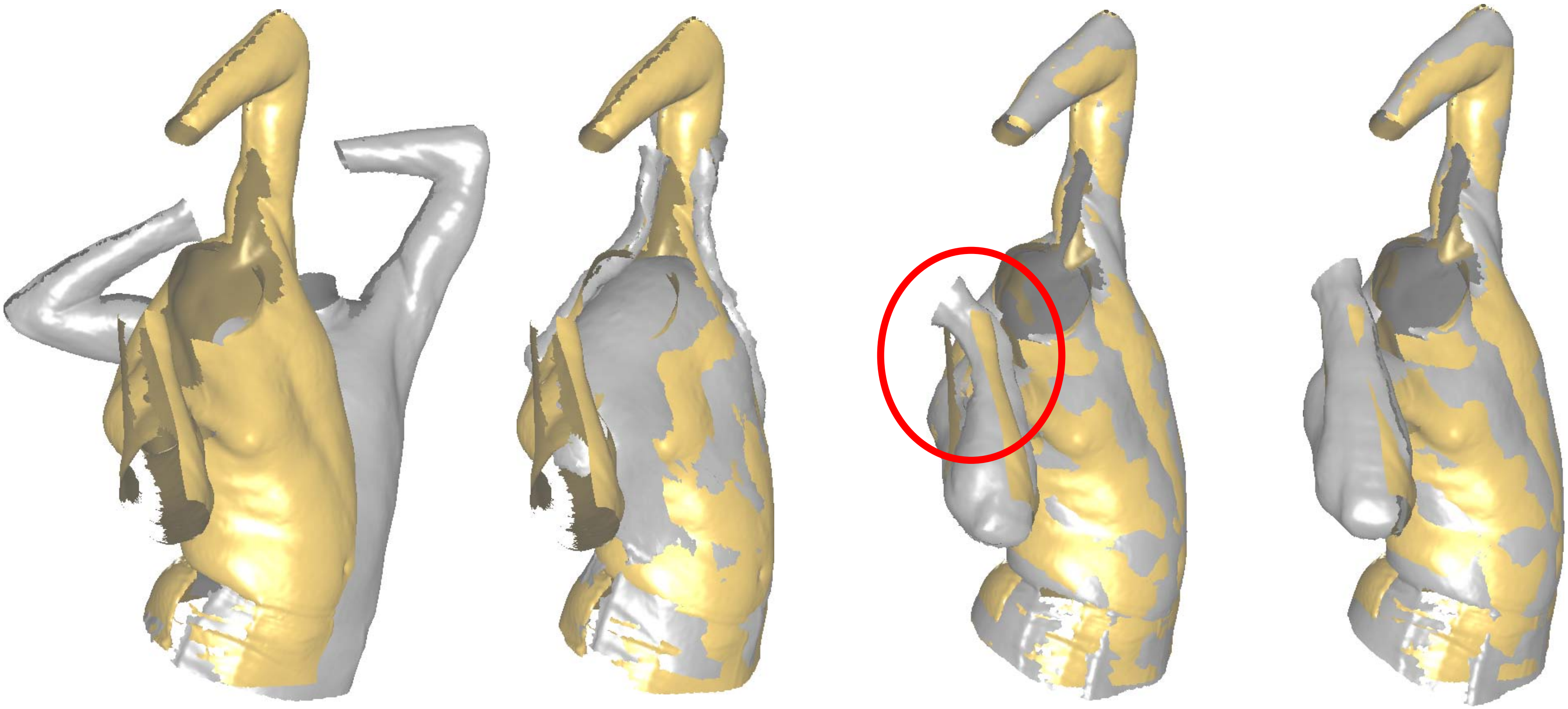
$$E_{\text{rigid}} = \sum_k E_k = \sum_k \sum_{\mathbf{s}_i \in \tilde{C}_k} \|\mathbf{R}_k \mathbf{s}_i + \mathbf{T}_k - \mathbf{s}_i'\|^2$$

Clusters for local rigidity

- Initially each cluster contains a single sample point
- Every 10 iterations (of correspondence & deformation), combine clusters that have similar rigid transformations (forming larger rigid parts)

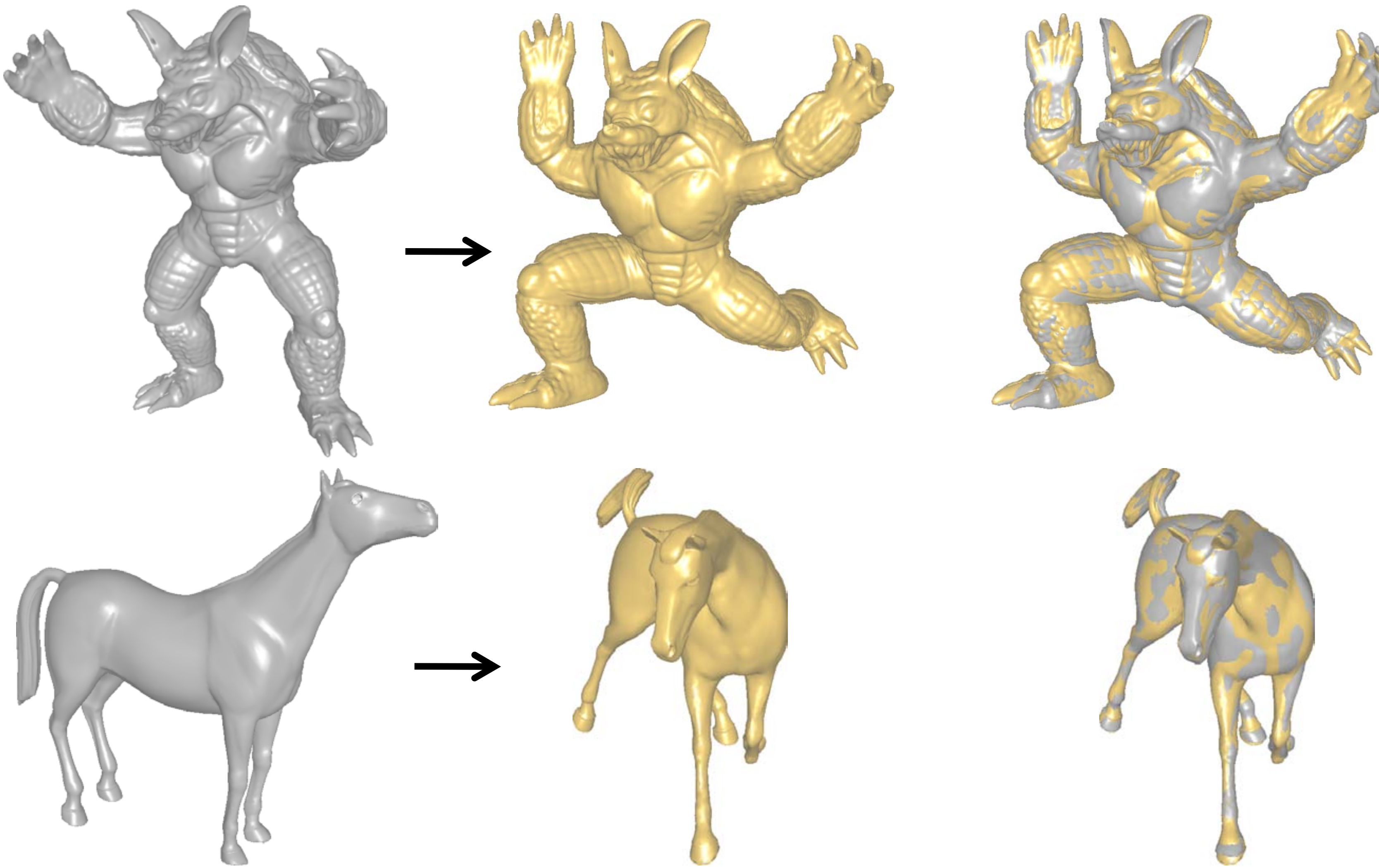


Advantages of features & clustering

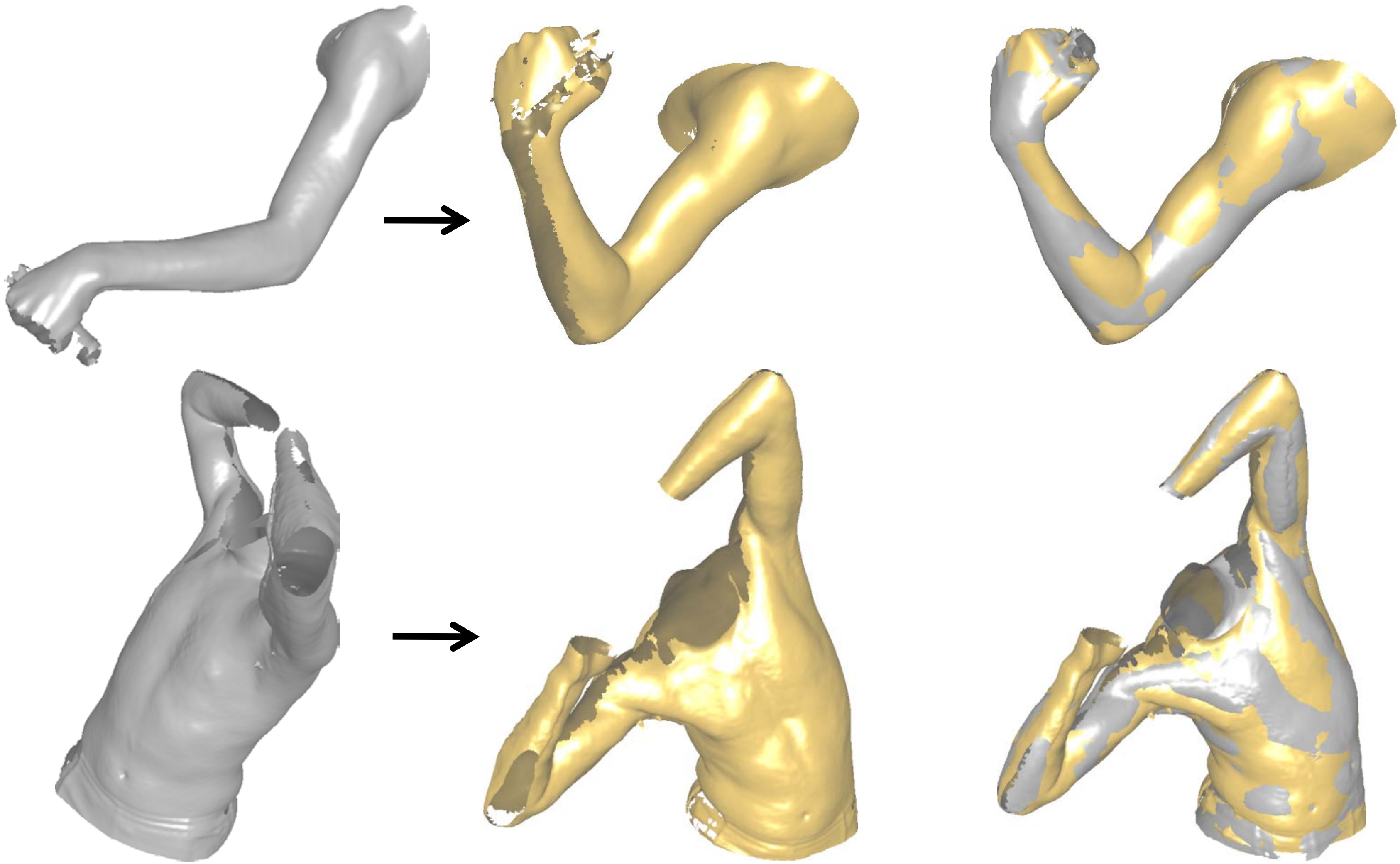


Source + Target **Without Features** **Without Clustering** **With Both**

Results

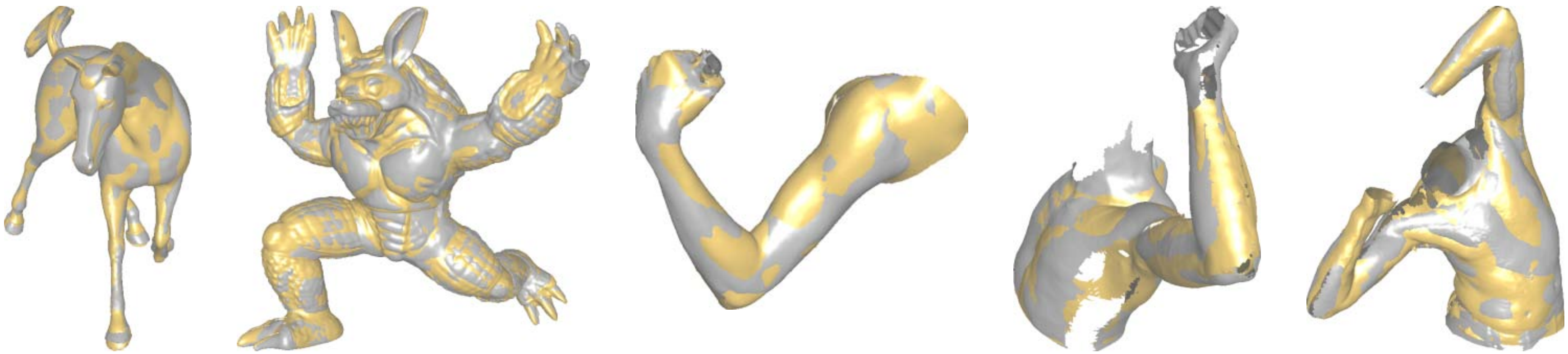


Results



Results

Efficient, robust method



data set	#poses	#pairs	$\lvert\mathcal{S}\rvert$	$\lvert\hat{\mathcal{S}}\rvert$	pre time	reg time
Horse	10	45	80k	2500	7.4s	13.6s
Armadillo	12	66	332k	2500	7.6s	14.8s
Arms	36	630	80k	600	2.1s	1.1s
Shoulder	33	528	117k	800	3.4s	1.9s
Torso	27	231	325k	1100	4.5s	4.5s

Conclusion

Correlated correspondence

- Robust method for matching correspondences
- Measure how much the correspondence "makes sense"
- Probability model → optimized using LBP
- Requires a template
  - If model is incomplete, then there is no "correct" corresponding point to assign

Conclusion

Non-rigid registration under isometric deformations

- Improve closest point correspondences using features and spectral matching
- Deform shape while preserving local rigidity of clusters
- Iteratively estimate correspondences and deformation until convergence
- Robust, efficient method
- Relies on geodesic distances (problematic when holes are too large)