Global Shape Matching

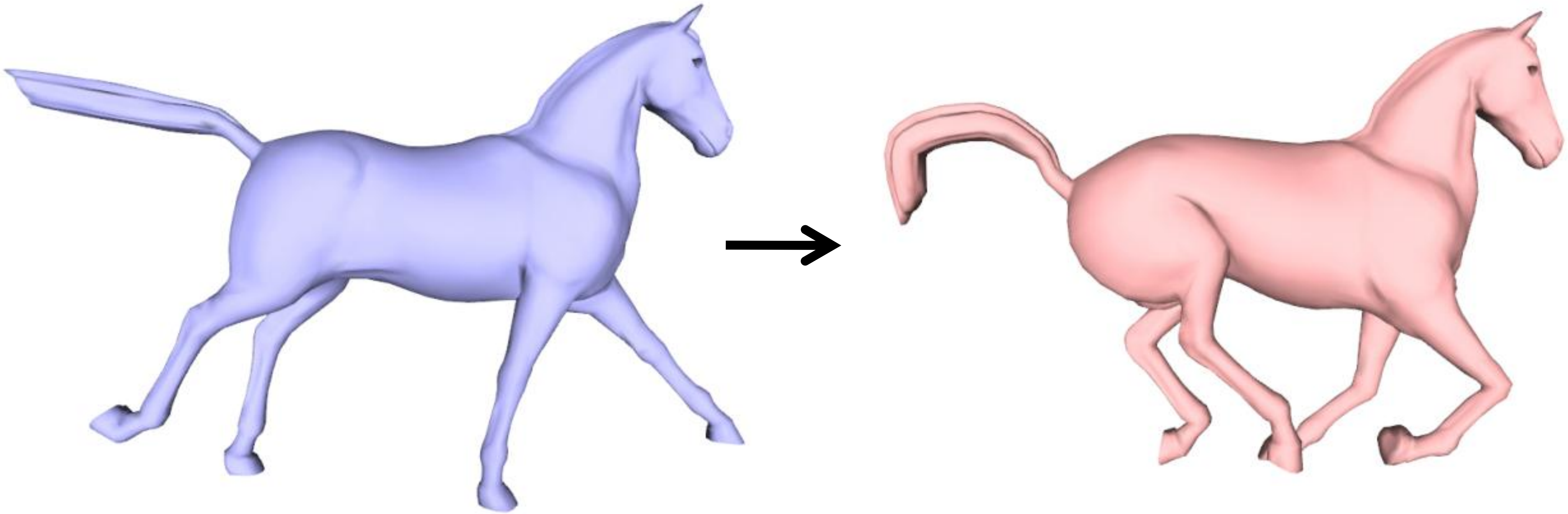
Section 3.3: Articulated Matching using Graph Cuts

Articulated Shape Matching

Feature-based matching alone is not enough to find correspondences

- Good for narrowing down search space

In this section: Leverage this idea to perform articulated shape matching



Correspondence Problem Classification

How many meshes?

- **Two:** Pairwise registration
- More than two: multi-view registration

Initial registration available?

- Yes: Local optimization methods
- **No:** Global methods

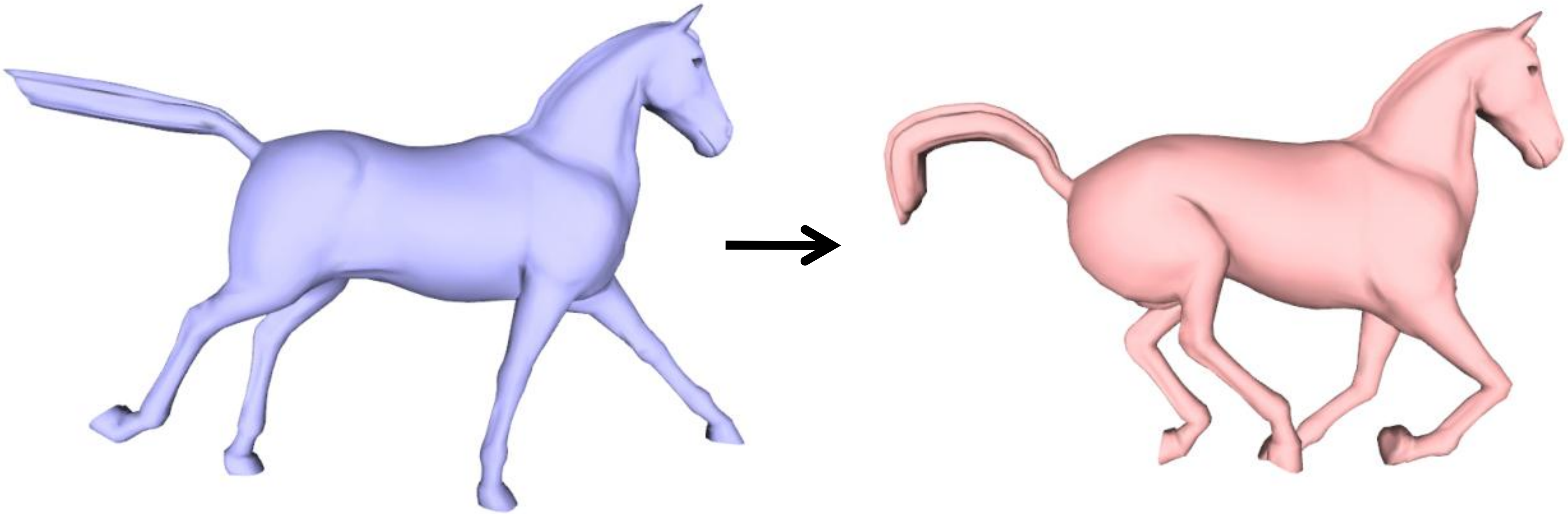
Class of transformations?

- **Rotation and translation:** Rigid-body (multiple parts)
- Non-rigid deformations

Basic Idea

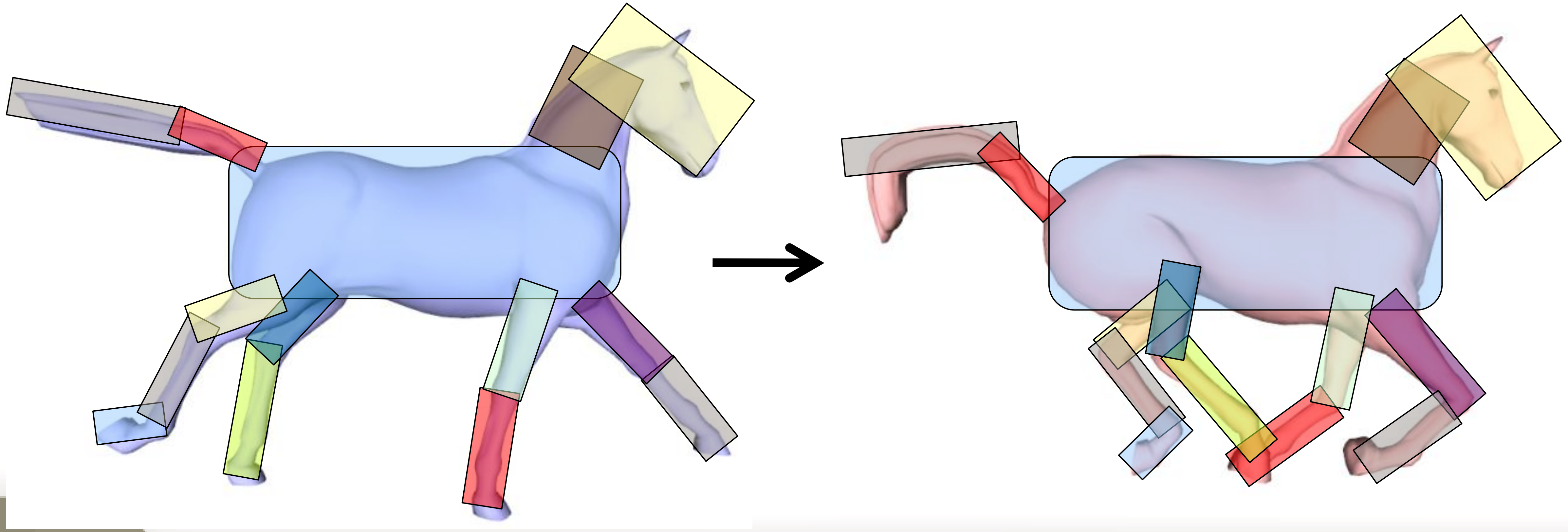
Two main steps

1. **Motion Sampling:** Find small set of transformations describing surface movement
2. **Optimization:** Figure out where to apply which transformation so that the surfaces match



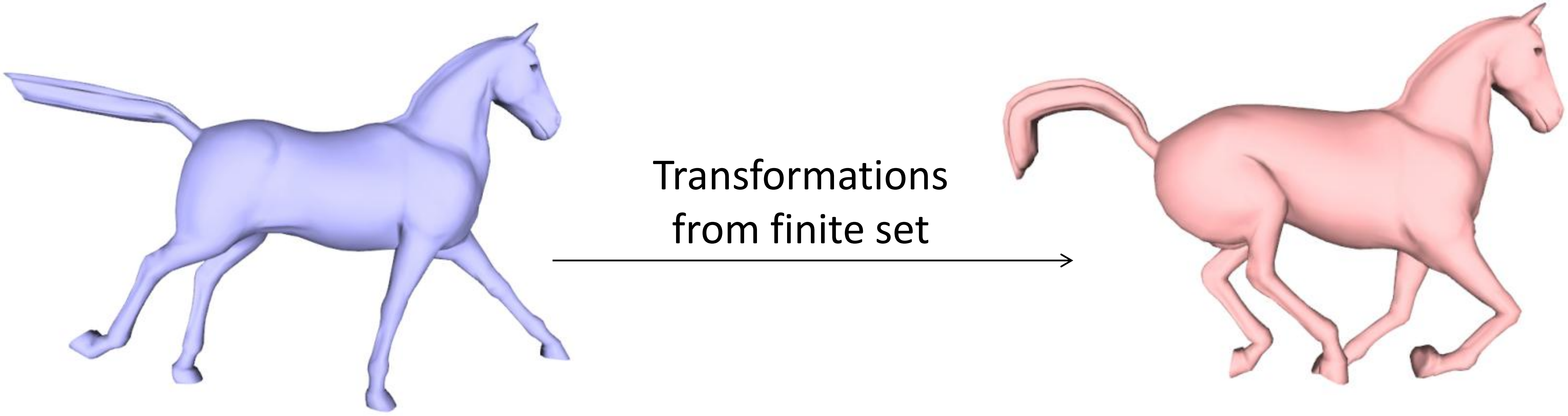
Basic Idea: Motion Sampling

- Each feature match guesses how that point moved
  - **Each match = a rigid transformation candidate**
- Property of articulated shapes: each rigid part moves according to a **single** rigid transformation
- Many transformation candidates will be the same!
  - Use voting scheme to group similar transformations



Basic Idea: Optimization

- If we know the movement of each part (i.e. extract set of transformations **{T}**)
- Find an assignment of transformations to the points that "minimizes registration error"



Source Shape P

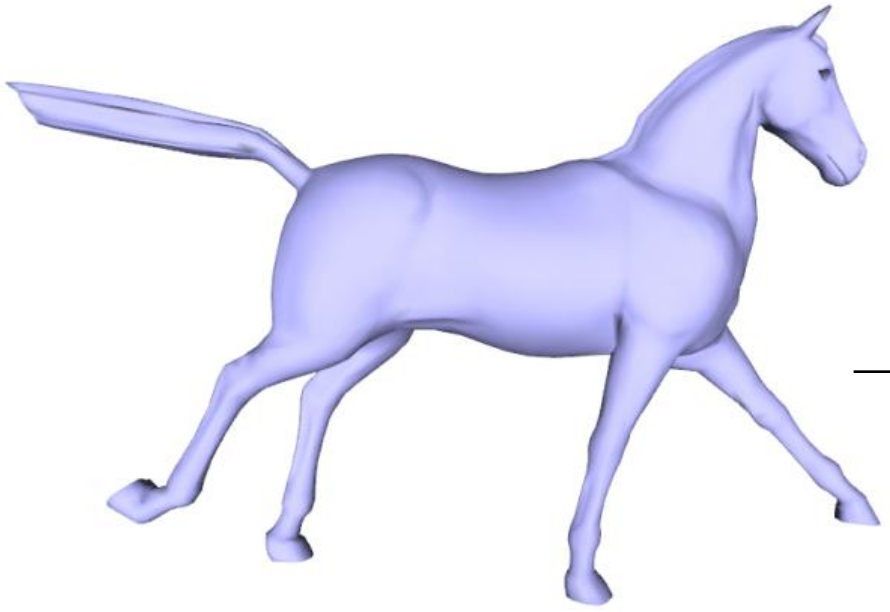
Target Shape Q

Basic Idea: Optimization

Find the assignment of transformations in {T} to points in P, that maximizes:

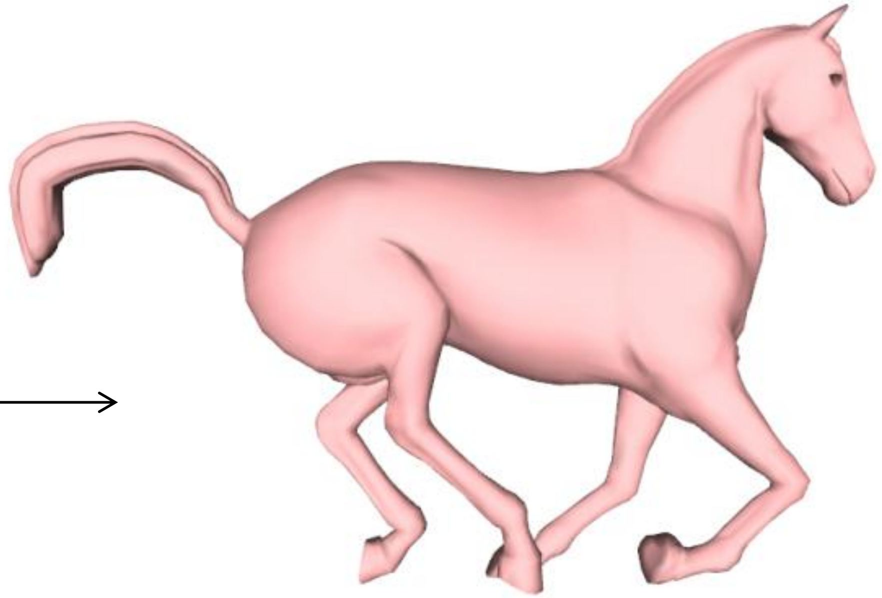
$$P^{(match)}(x_1,\dots,x_n)=\prod_{i=1}^{n}P_i^{(single)}\prod_{i,j=1}^{n}P_{i,j}^{(compatible)},\ x_i\in\{T\}$$

“Data” and “Smoothness” terms evaluate quality of assignment



Source Shape P

Transformations from finite set $\longrightarrow$



Target Shape Q

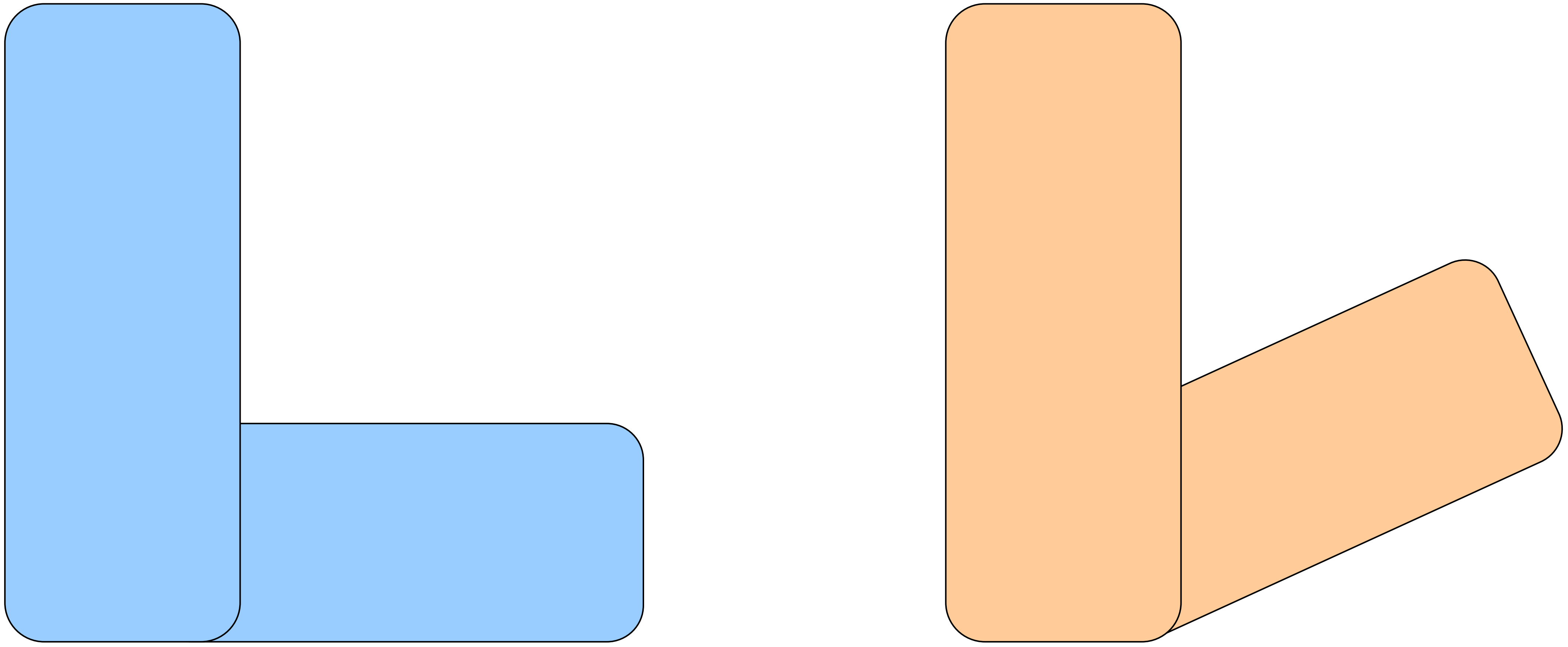
How to find transformations?

Global search / feature matching strategy [CZ08]

- Sample transformations in advance by feature matching
- Inspired by partial symmetry detection [MGP06]
  - Covered later in the course!

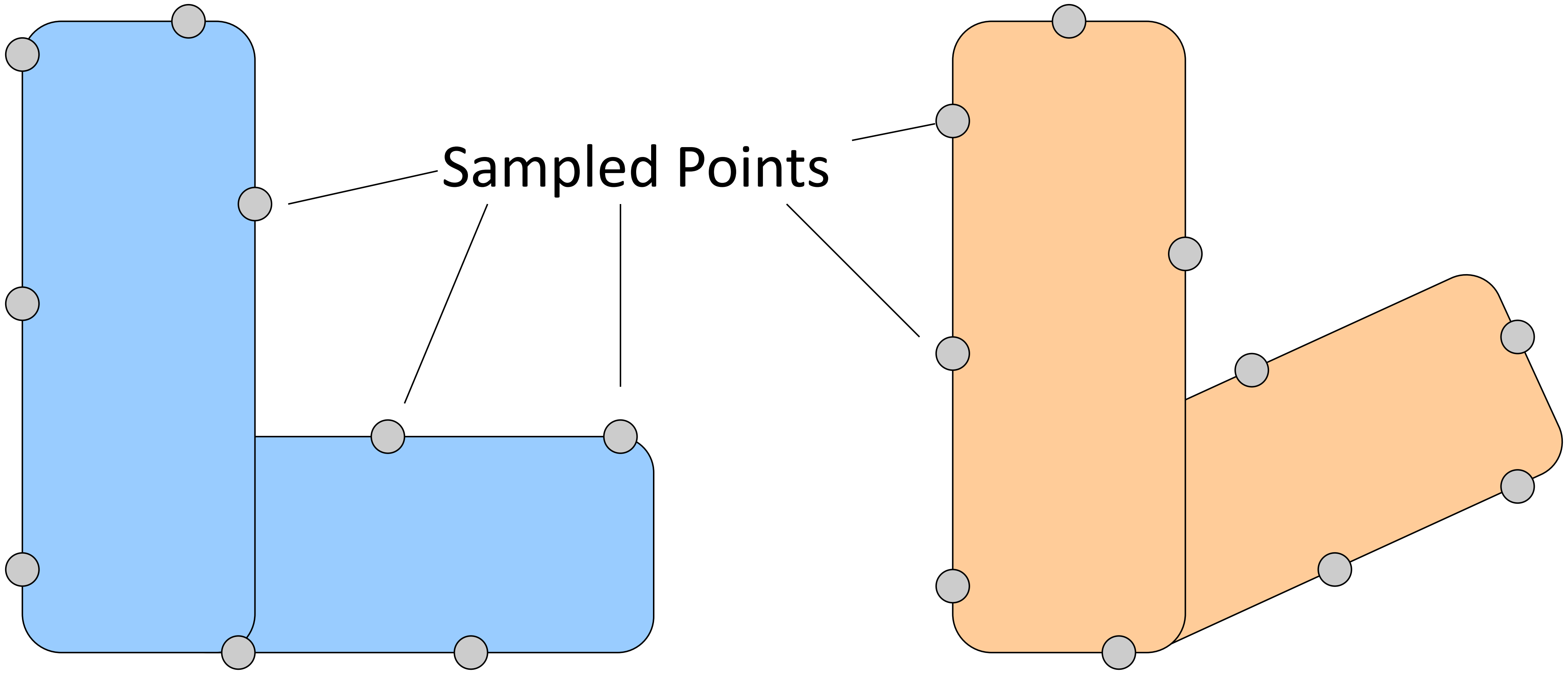
Motion Sampling Illustration

Find transformations that move parts of the source to parts of the target



Motion Sampling Illustration

Find transformations that move parts of the source to parts of the target

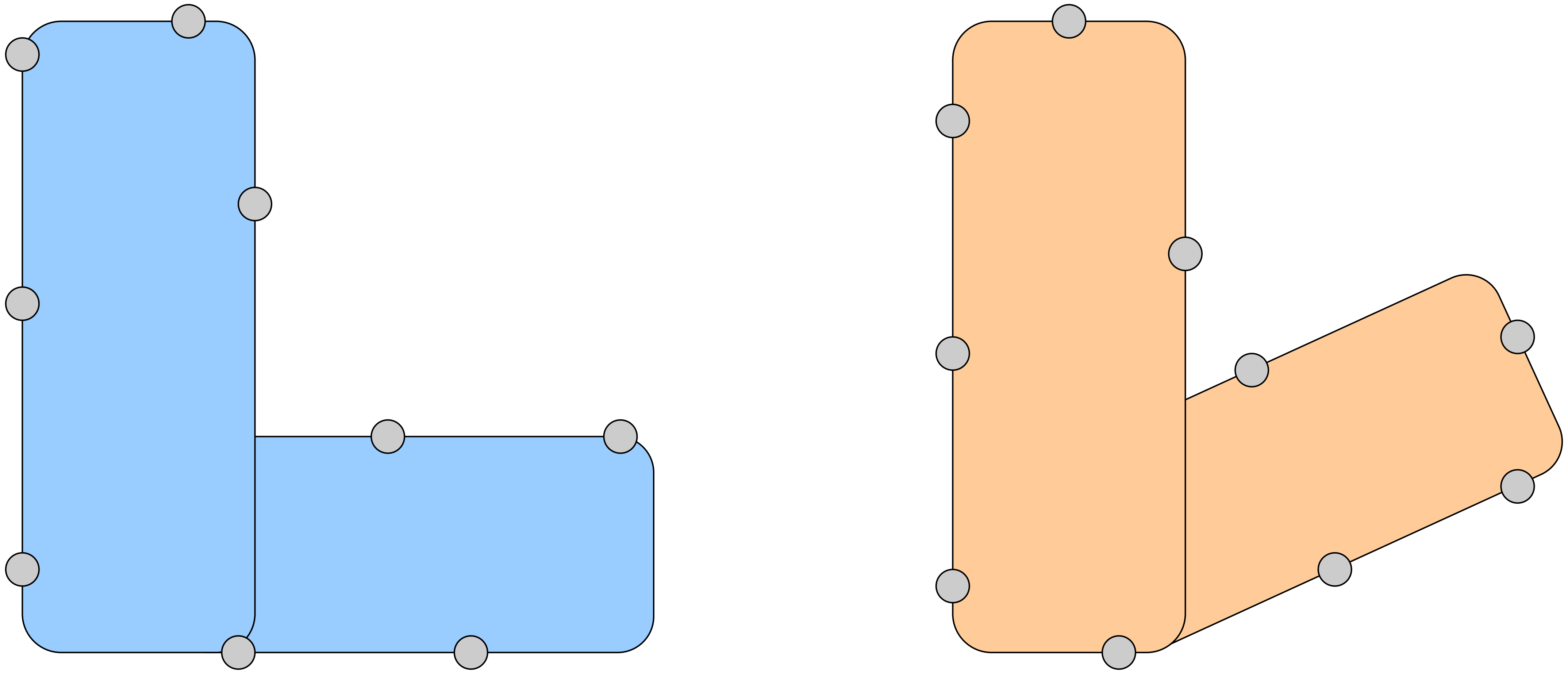


Source Shape

Target Shape

Motion Sampling Illustration

- Find transformations that move parts of the source to parts of the target

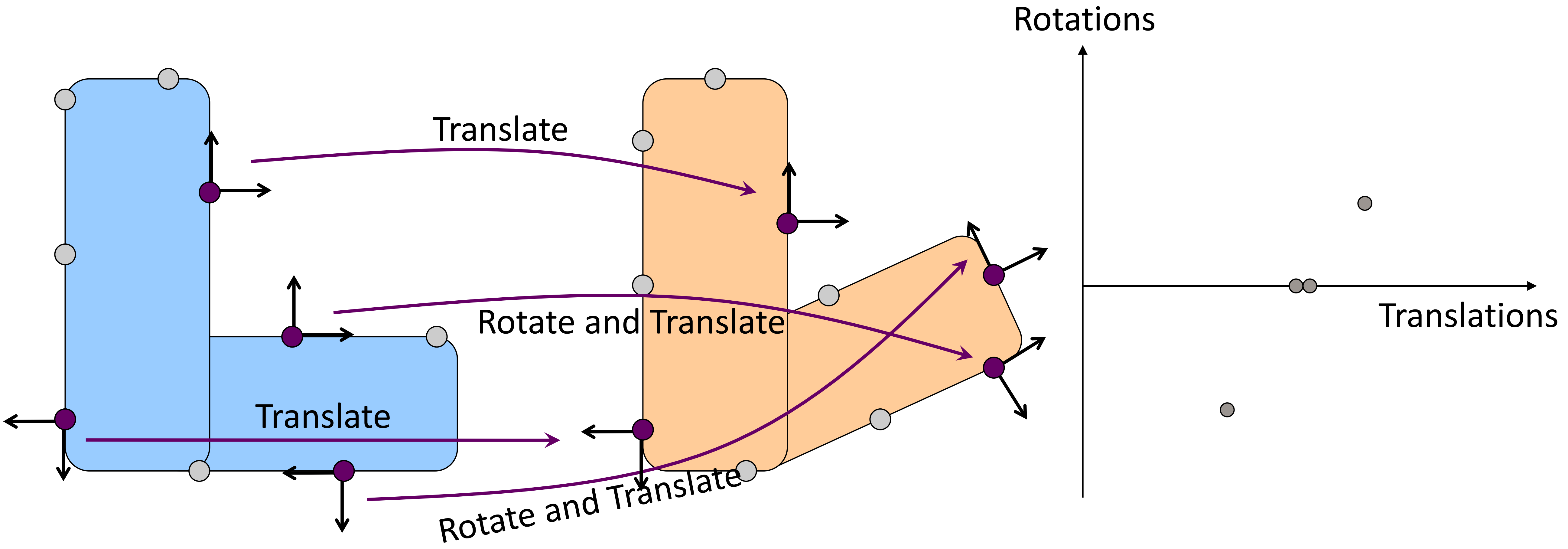


Source Shape

Target Shape

Motion Sampling Illustration

Find transformations that move parts of the source to parts of the target



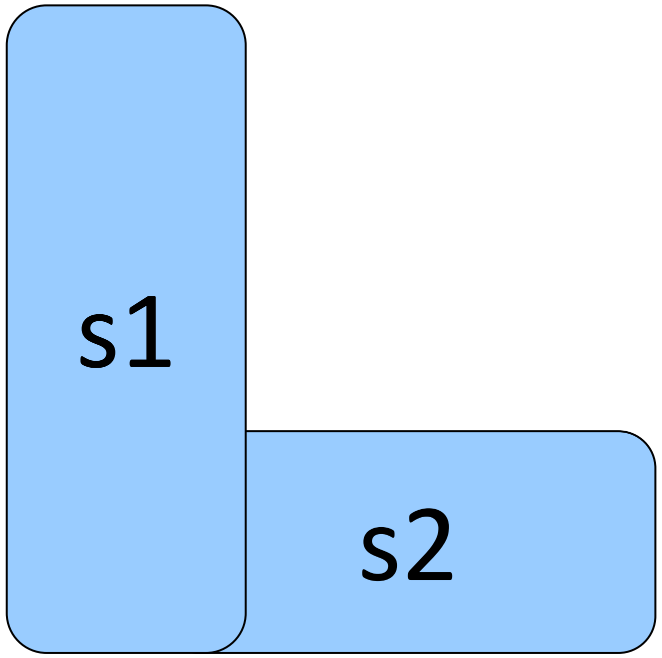
Source Shape

Target Shape

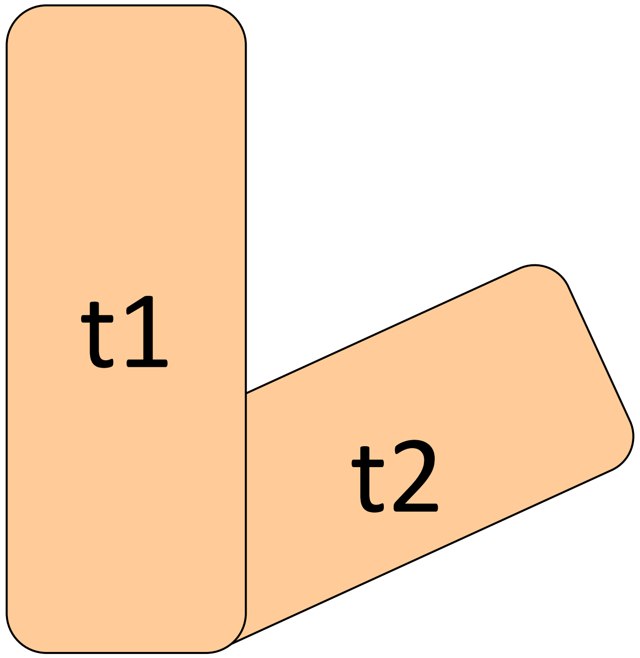
Transformation Space

Motion Sampling Illustration

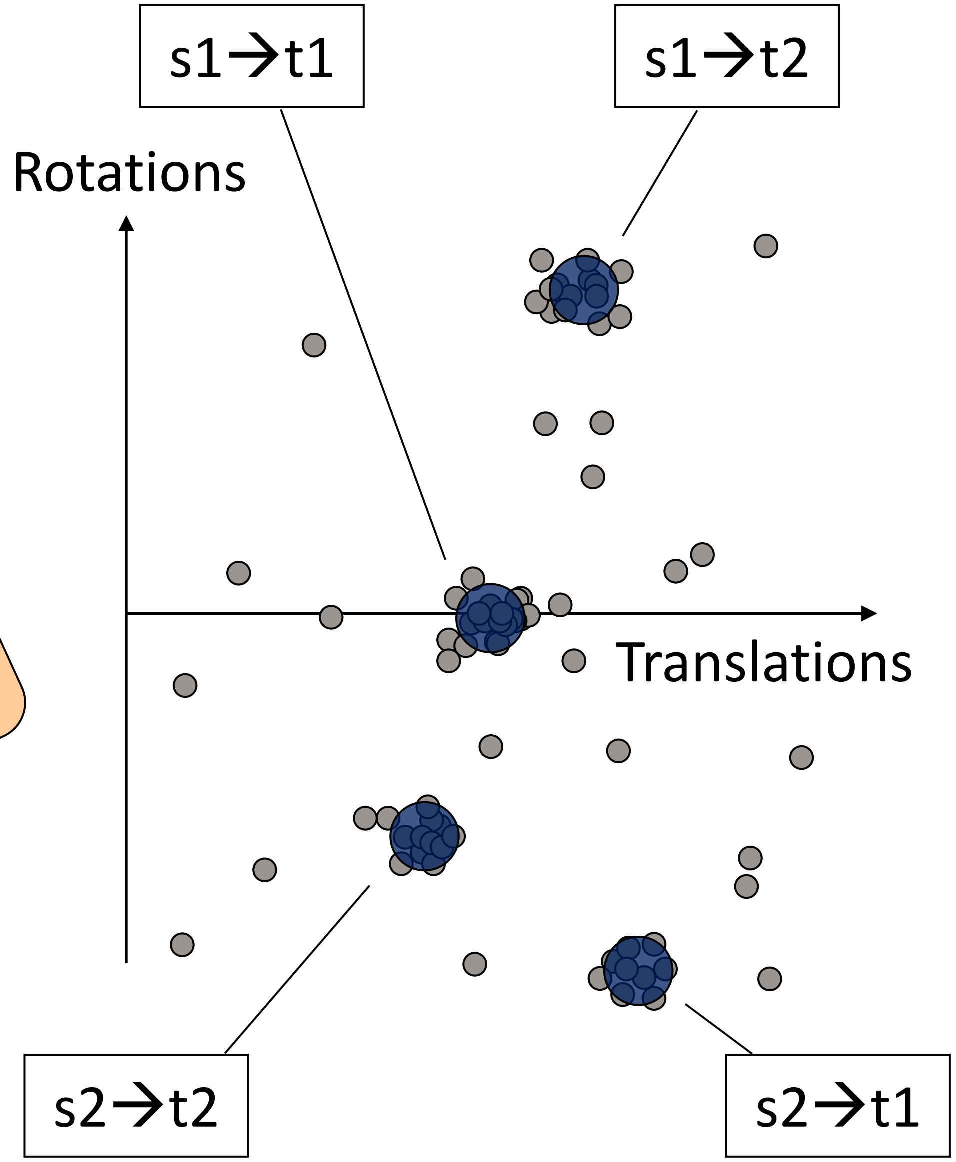
Find transformations that move parts of the source to parts of the target



Source Shape



Target Shape



Transformation Space

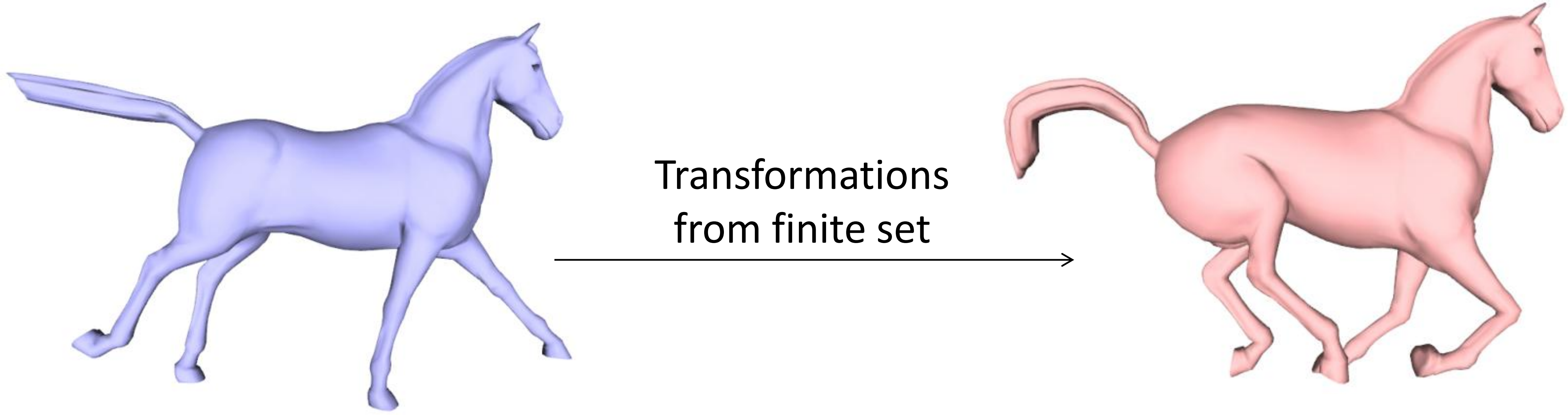
Basic idea

Find the assignment of transformations in {T} to points in P, that maximizes:

$$P^{(match)}(x_1,\ldots,x_n)=\prod_{i=1}^{n}P_i^{(single)}\prod_{i,j=1}^{n}P_{i,j}^{(compatible)},\ x_i\in\{T\}$$

"Data" and "Smoothness" terms evaluate quality of assignment

A *discrete labelling problem* → Graph Cuts for optimization



Source Shape P

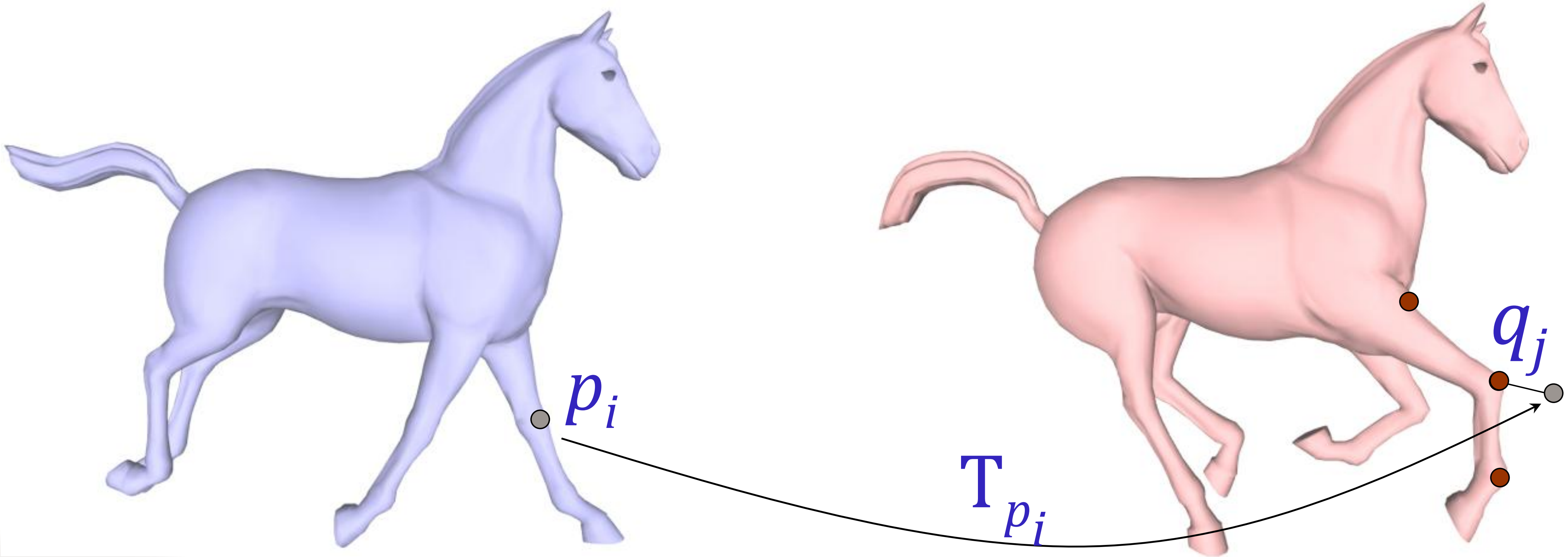
Target Shape Q

Data Term

For each mesh vertex: Move close to target

How to measure distance to target?

- Apply assigned transformation T_{p_i} for all $p_i \in P$
- Measure distance to closest point q_j in target

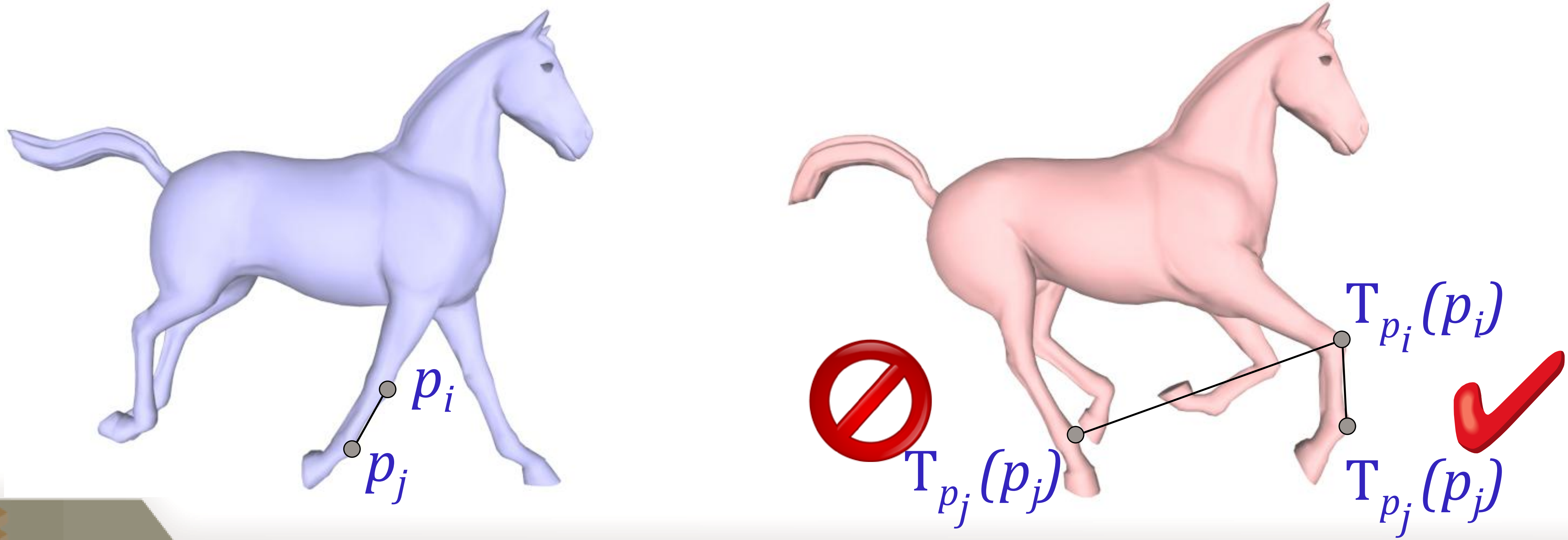


Smoothness Term

For each mesh edge: preserve length of edge

$$V(p_i, p_j, T_{p_i}, T_{p_j}) = \left| \underbrace{\|p_i - p_j\|}_{\text{Original Length}} - \underbrace{\| T_{p_i}(p_i) - T_{p_j}(p_j) \|}_{\text{Transformed Length}} \right|$$

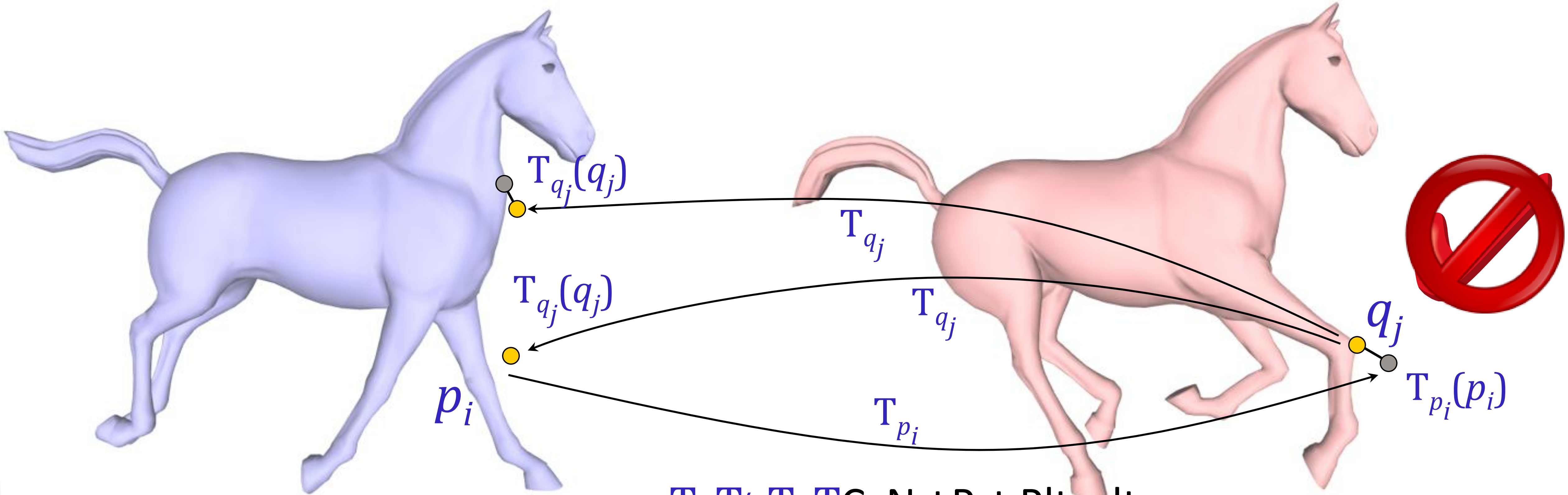
- Both versions of $T_{p_j}(p_j)$ moved p_j close to the target
- Disambiguate by preferring the one that preserves length



Symmetric Cost Function

Swapping source / target can give different results

- Optimize $\{T\}$ assignment in both meshes
- Assign $\{T\}$ on source vertices, $\{T^{-1}\}$ on target vertices
- Enforce consistent assignment: penalty when $T_{p_i} \neq T_{q_j}$



$T_{p_i} \neq T_{q_j}$ Constant Penalty

$T_{p_i} = T_{q_j}$ No Penalty

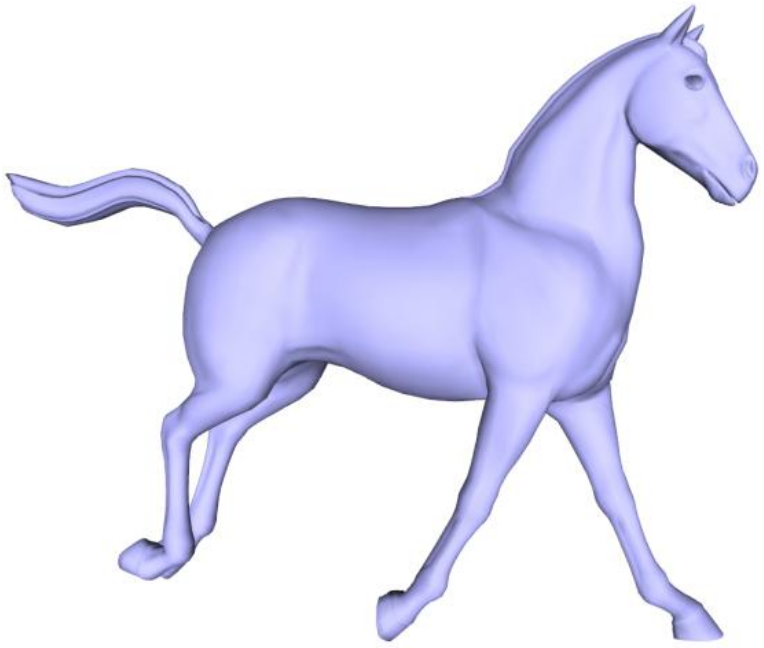
Optimization Using Graph Cuts

argmin
Assignment from a set of transformations

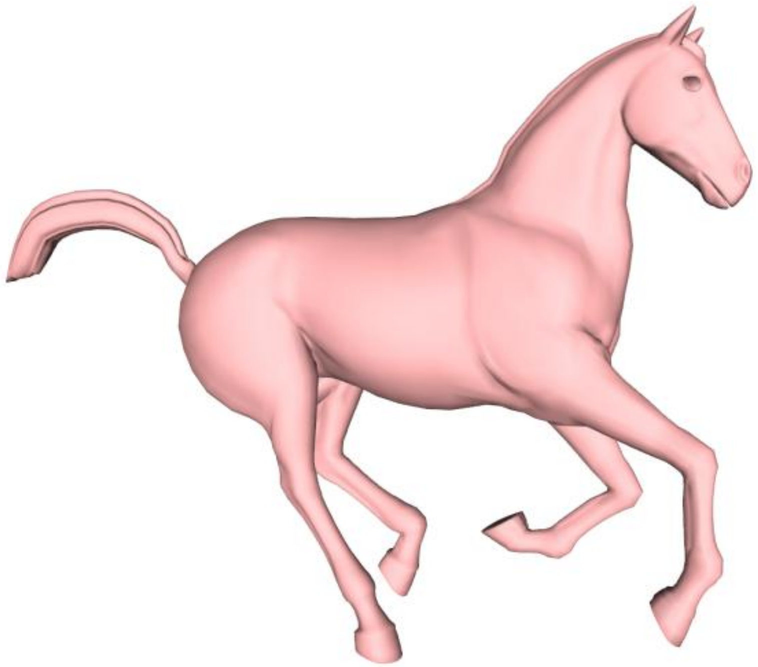
$$\text{Data}_{\textit{Source}} + \text{Smoothness}_{\textit{Source}} + \text{Data}_{\textit{Target}} + \text{Smoothness}_{\textit{Target}} + \text{Symmetric Consistency}_{\textit{Source \& Target}}$$

- Data and smoothness terms apply to both shapes
    - Additional symmetric consistency term
- Weights to control relative influence of each term
- Use “graph cuts” to optimize assignment
    - [Boykov, Veksler & Zabih PAMI ’01]

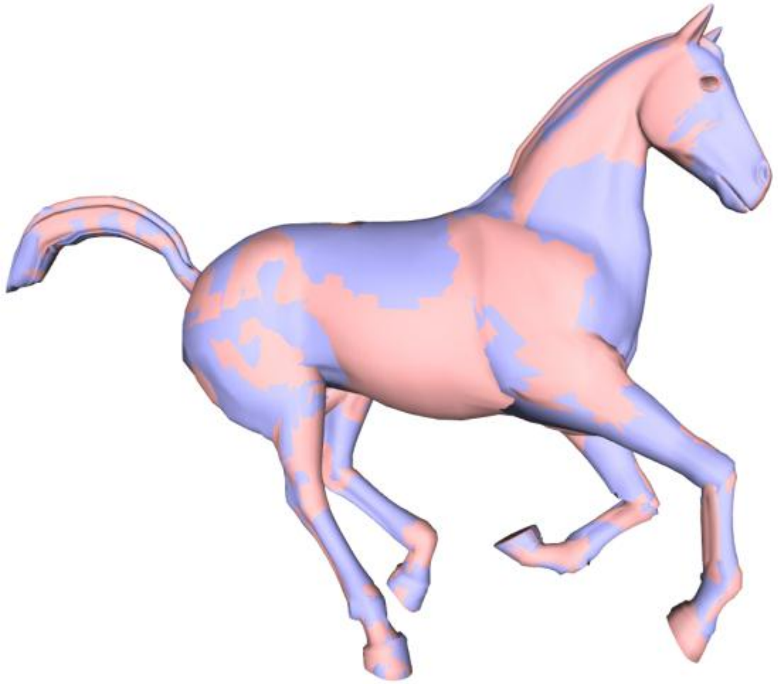
Synthetic Dataset Example



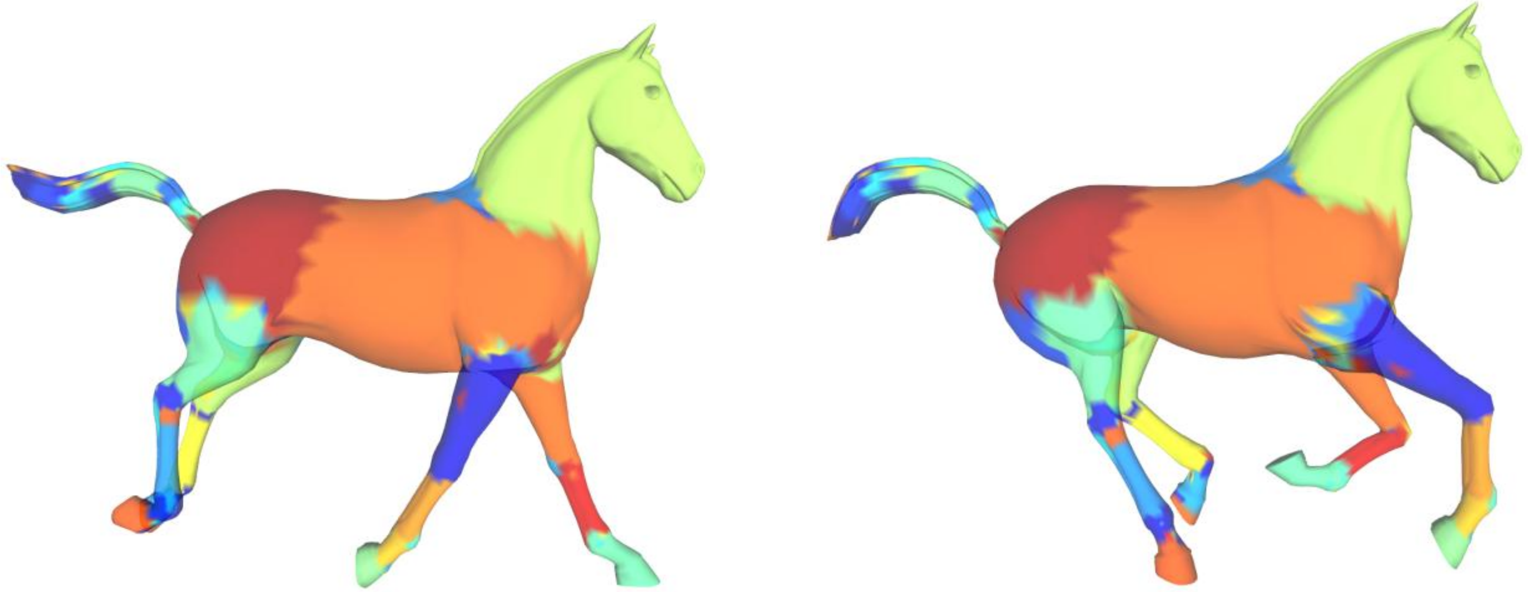
Source



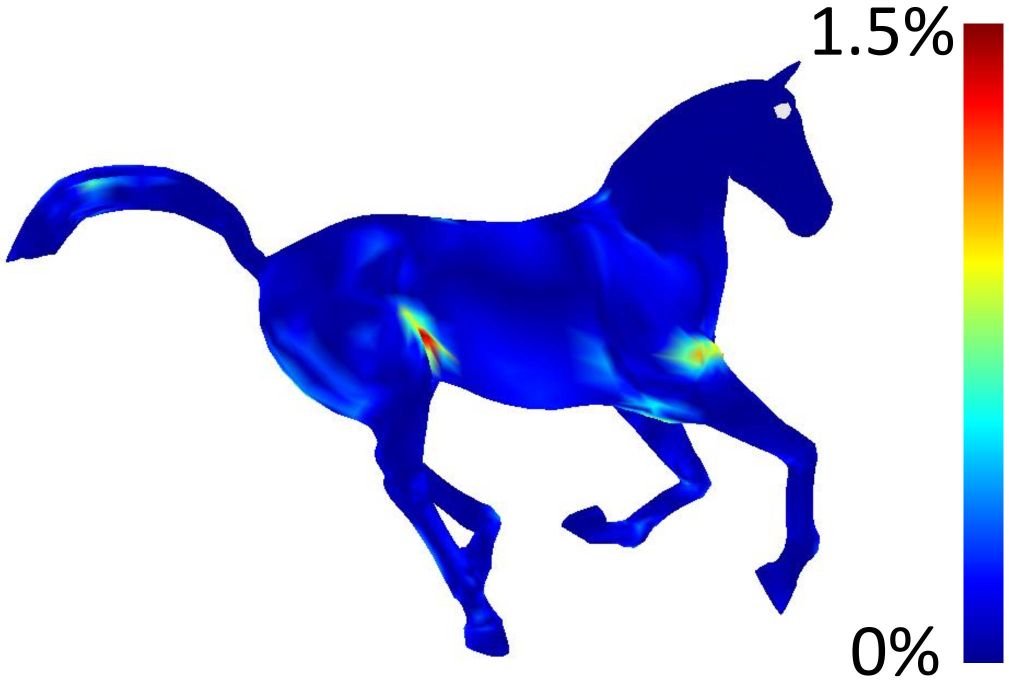
Target



Aligned Result

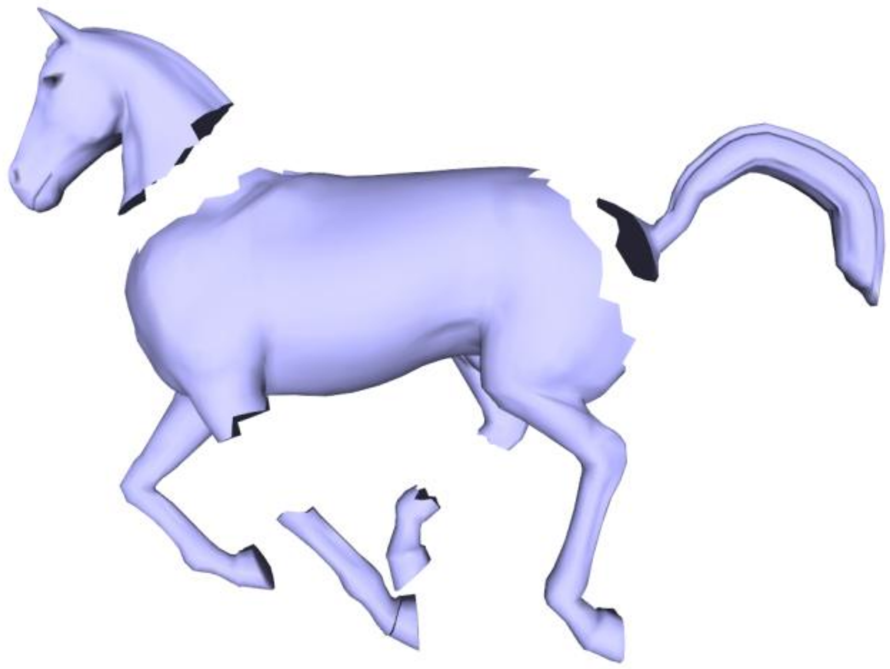


Motion Segmentation (from Graph Cuts)

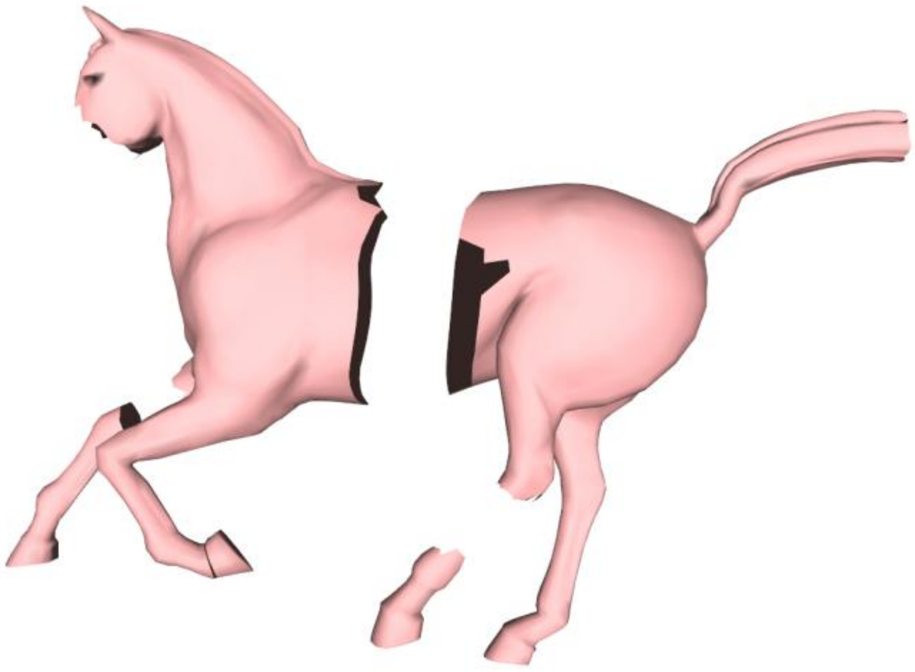


Registration Error

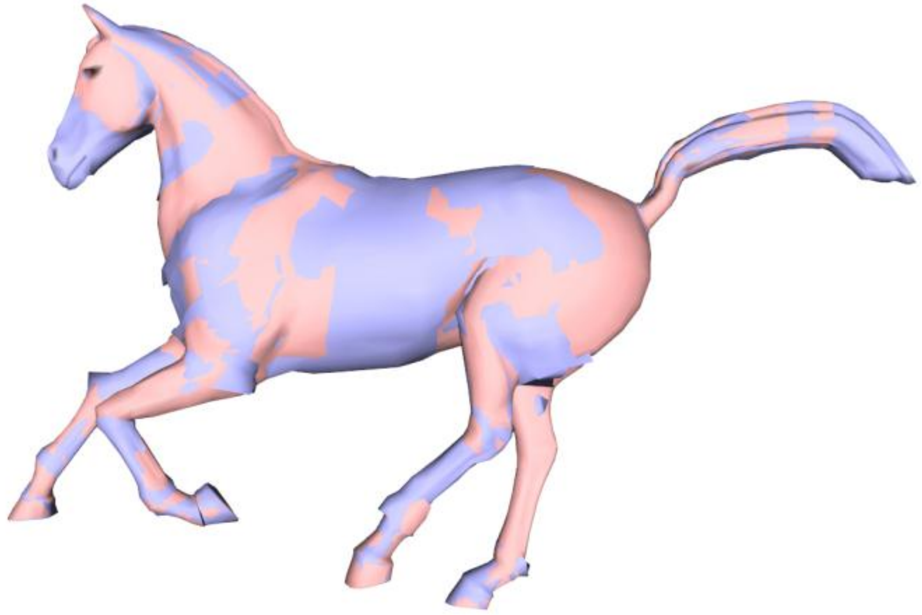
Synthetic Dataset w/ Holes



Source



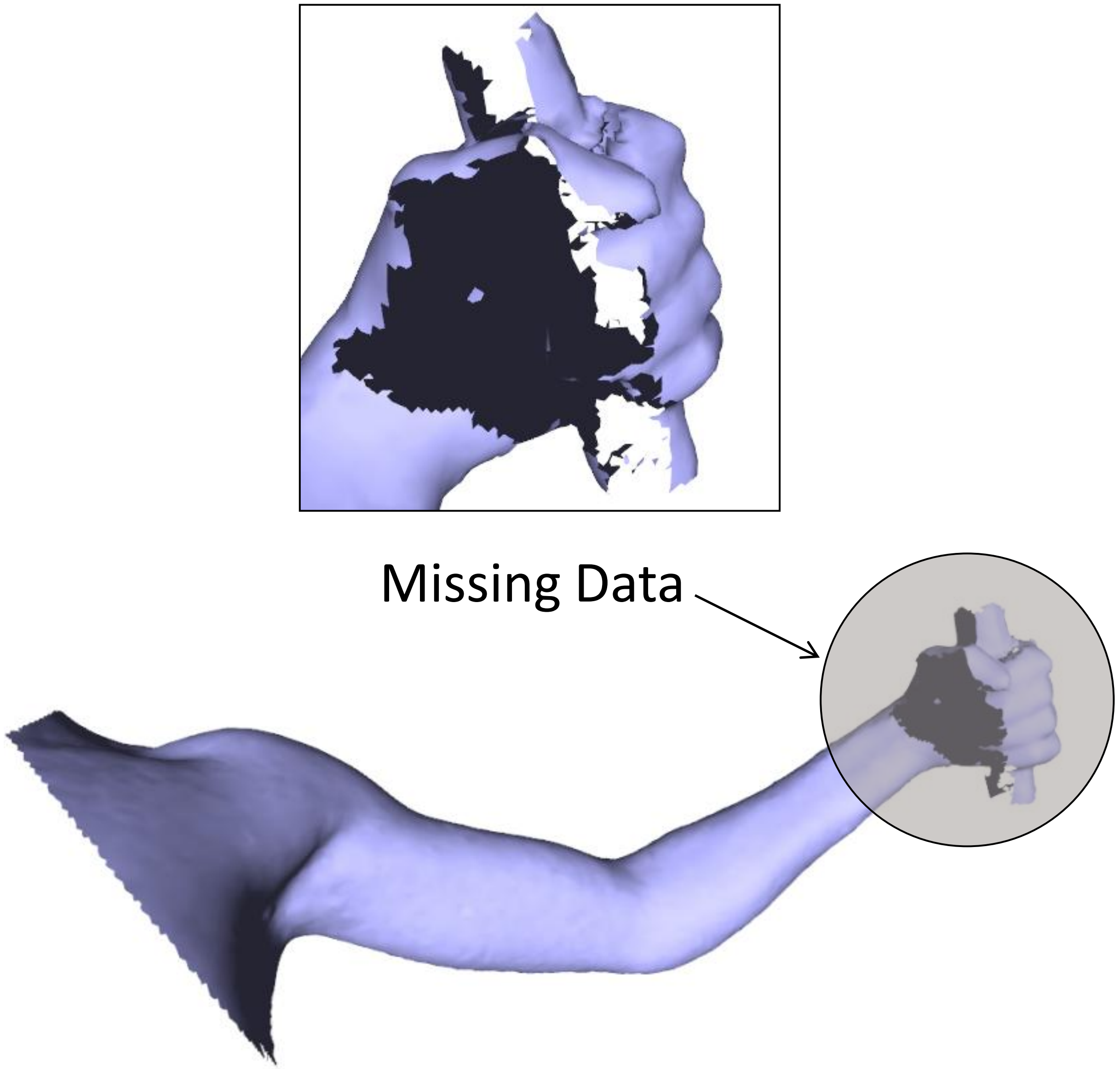
Target



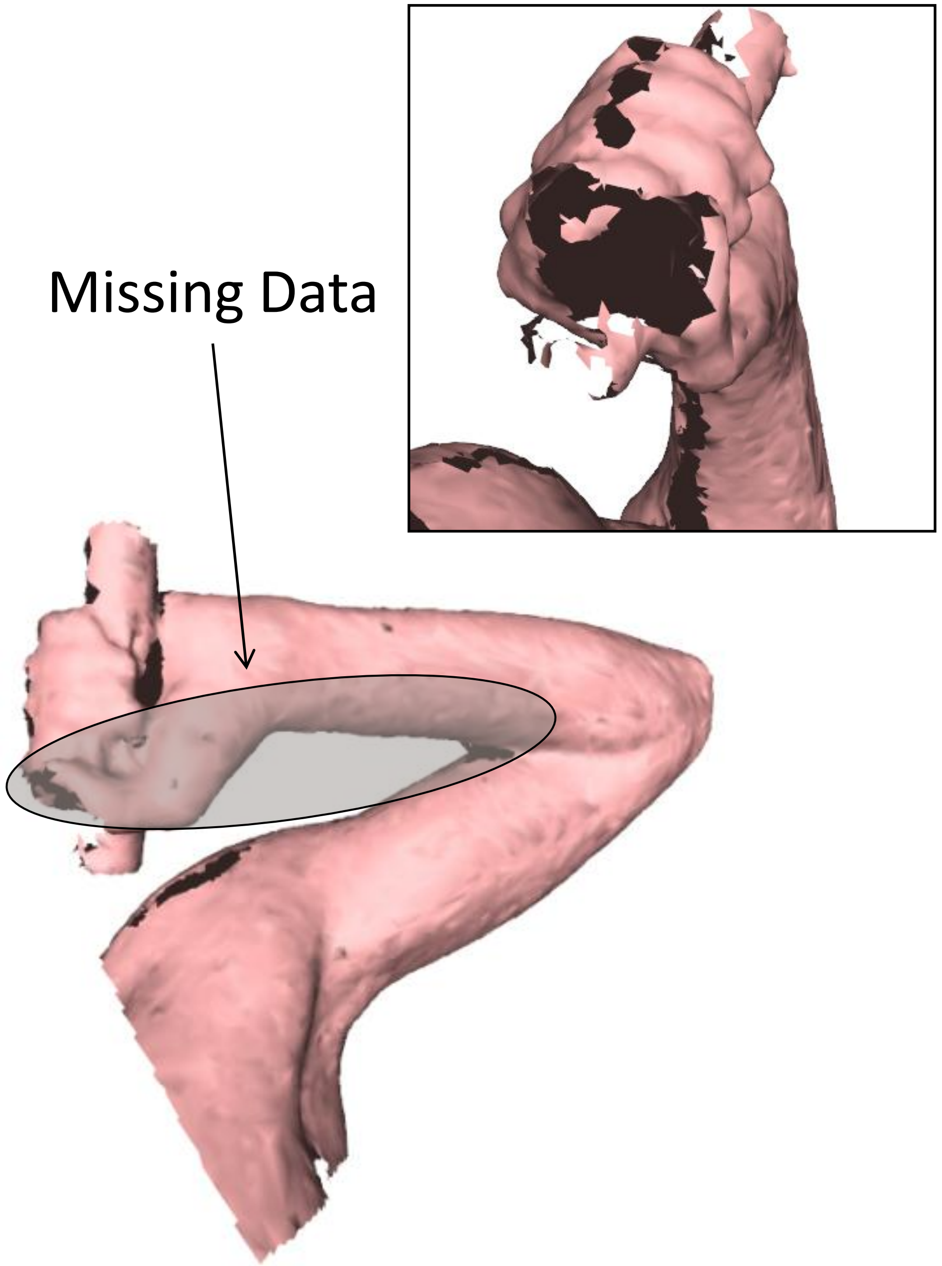
Aligned Result

Distance (from Target) to the closest point
(% bounding box diagonal)

Arm Dataset Example

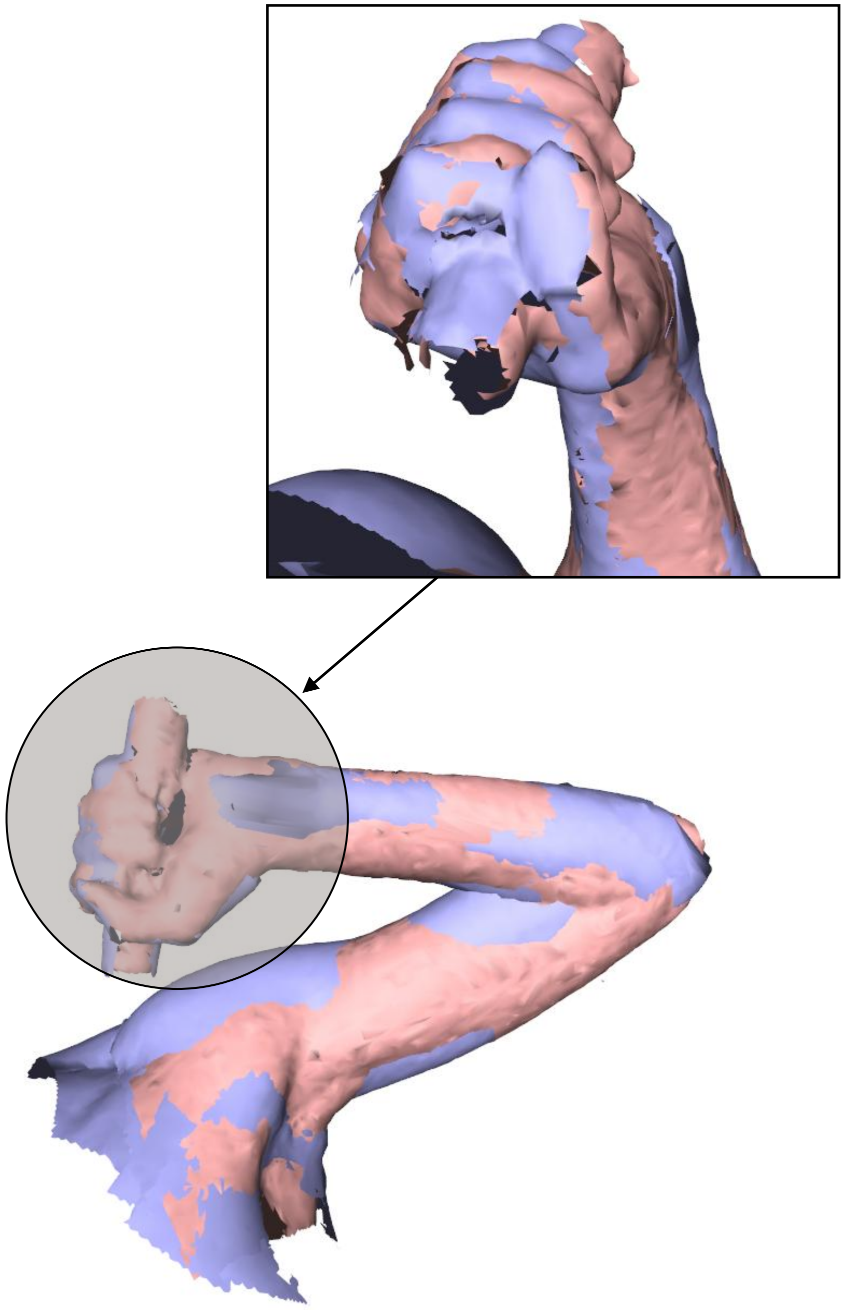


Source

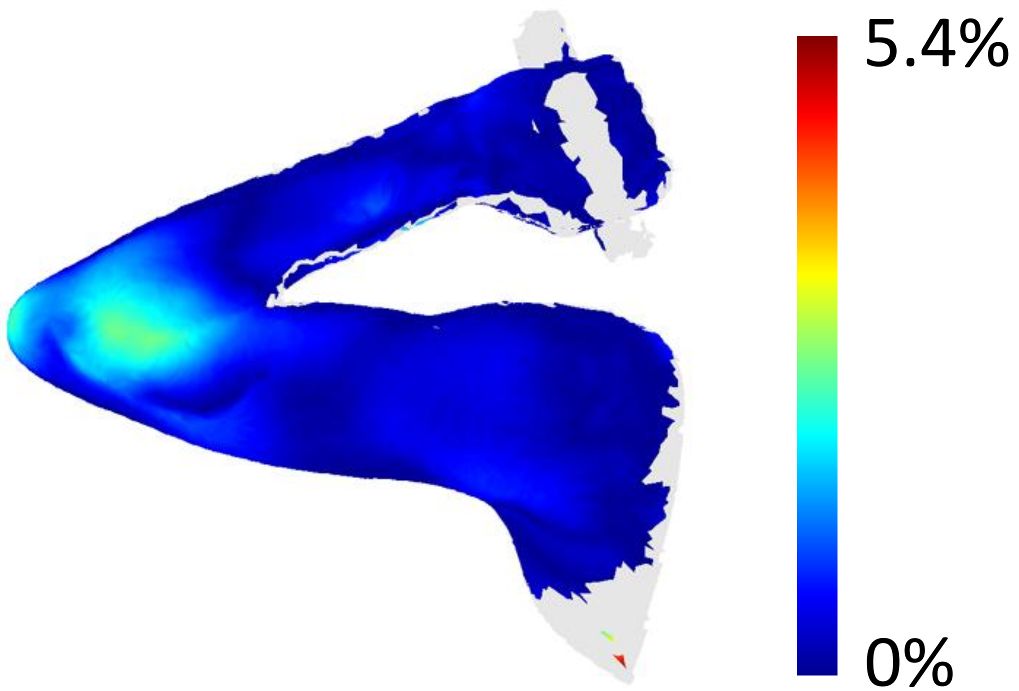


Noisy Target

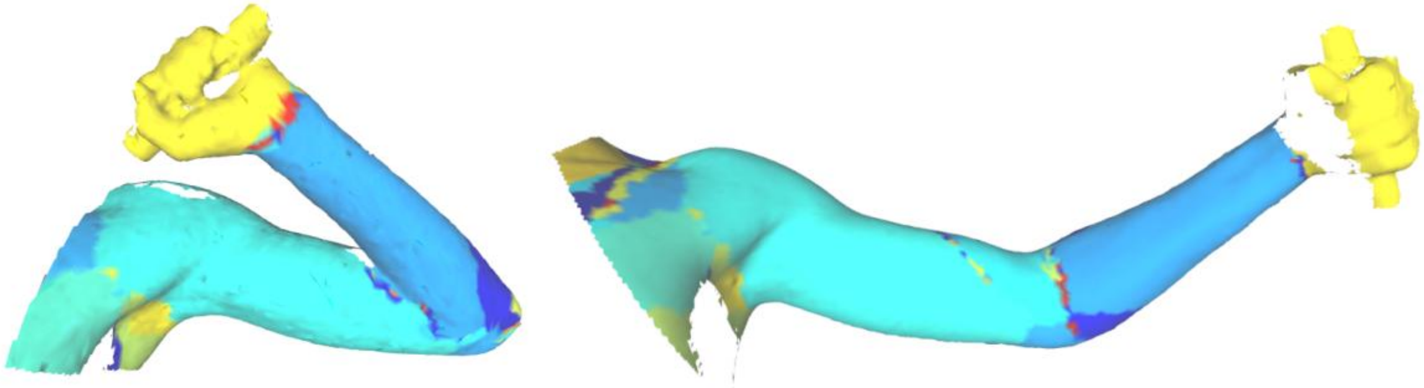
Arm Dataset Example



Aligned Result



Distance (from Target) to the closest point (% bounding box diagonal)



Motion Segmentation

Performance

Dataset	#Points	# Labels	Matching	Clustering	Pruning	Graph Cuts
Horse	8431	1500	2.1 min	3.0 sec	(skip) 1.6 sec	1.1 hr
Arm	11865	1000	55.0 sec	0.9 sec	12.4 min	1.2 hr
Hand (Front)	8339	1500	14.5 sec	0.7 sec	7.4 min	1.2 hr
Hand (Back)	6773	1500	17.3 sec	0.9 sec	9.4 min	1.6 hr

Graph cuts optimization is most time-consuming step

- Symmetric optimization doubles variable count
- Symmetric consistency term introduces many edges

Performance improved by subsampling

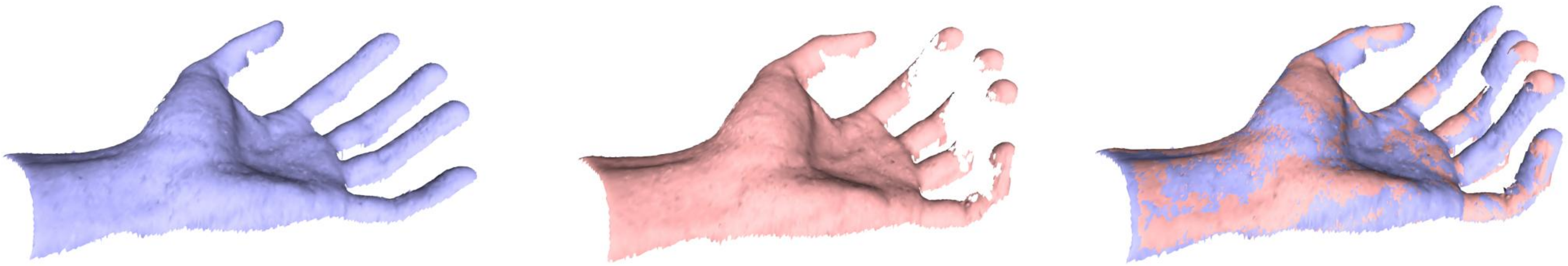
- Use k-nearest neighbors for connectivity

Pros/Cons

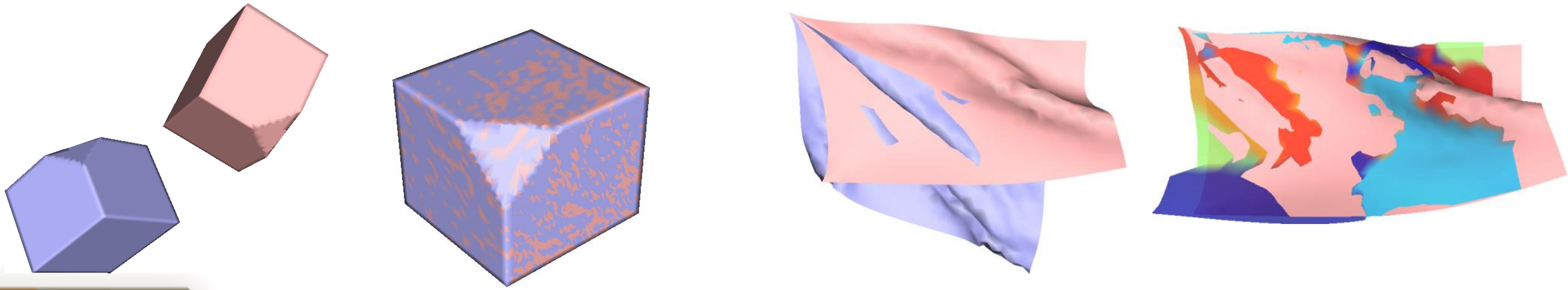
Pro: Feature matching is insensitive to initial pose

Con: *May fail to sample transformations properly when too much missing data / non-rigid motion*

Con: *Hard assignment of transformations*



Source Target Registration



Conclusions

Global shape matching for articulated shapes

- Features provide candidate transformations describing surface movement
- Optimize the assignment of transformations using graph cuts
- No marker, template, segmentation information needed
- Robust to occlusion & missing data

Thank you for listening!