Registration

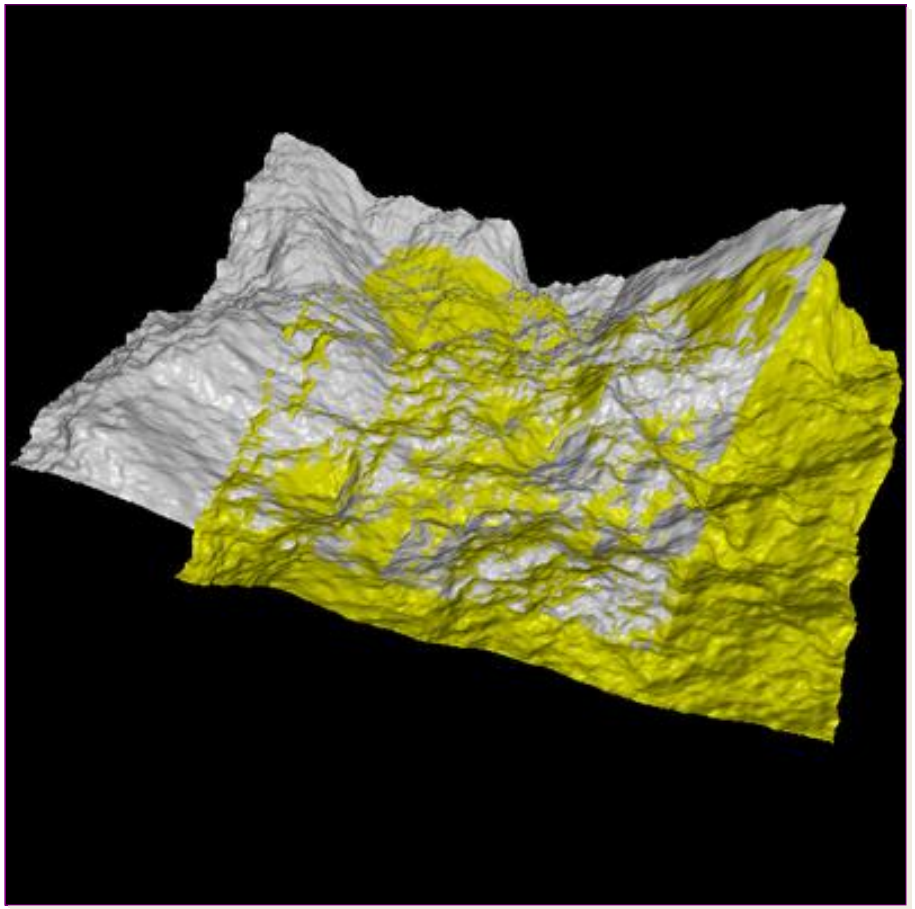
Umberto Castellani

Robotics Vision and Control

3D modelling from reality pipeline

Overall aim

Align partially-
overlapping views
given initial guess
for relative transform



Overall aim

Two views registration

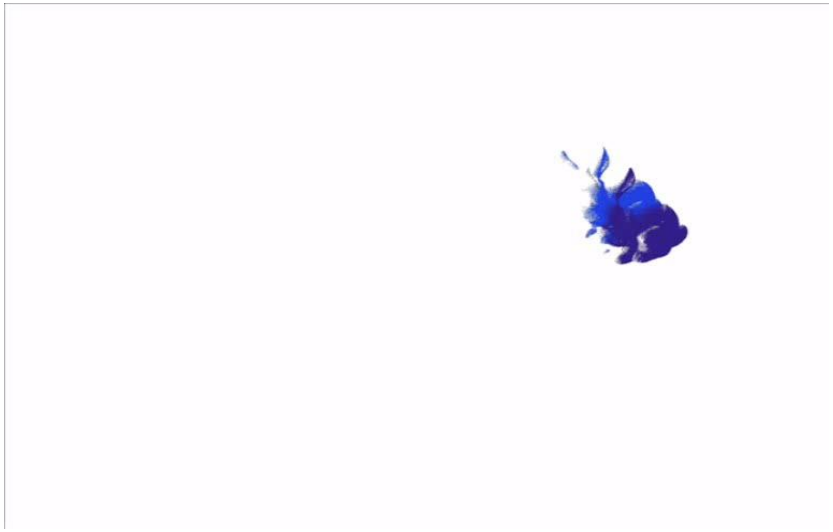
Gelfand, Mitra, Guibas, Pottmann, 2005

Overall aim

Multiple views registration

Gelfand, Mitra, Guibas, Pottmann, 2005

Overal aim

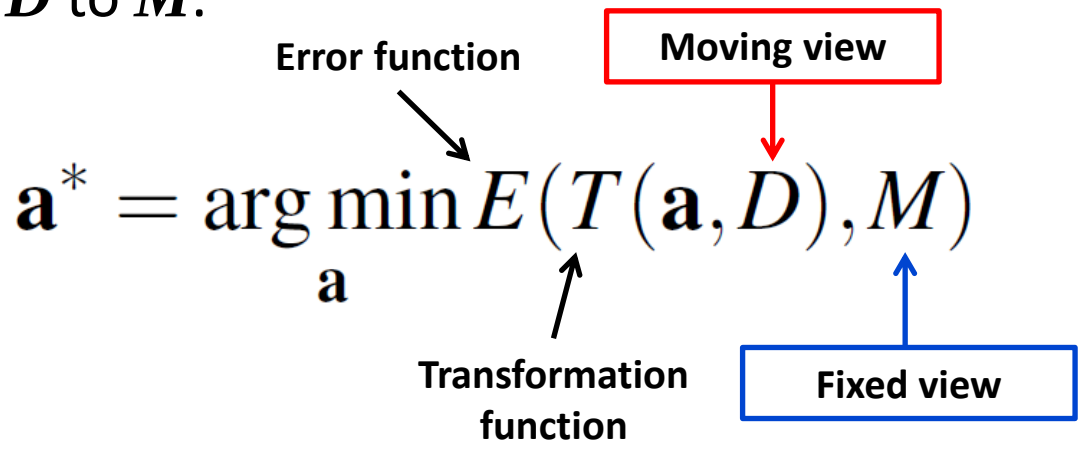


Multiple views

Two-view registration

Problem statement

Given a pair of views $\boldsymbol{D}$ and $\boldsymbol{M}$ representing two scans (partial 3D views) of the same object, **registration** is the problem of finding the parameters $\mathbf{a}^*$ of the transformation function $\boldsymbol{T}(\mathbf{a};\boldsymbol{D})$ which best aligns $\boldsymbol{D}$ to $\boldsymbol{M}$.



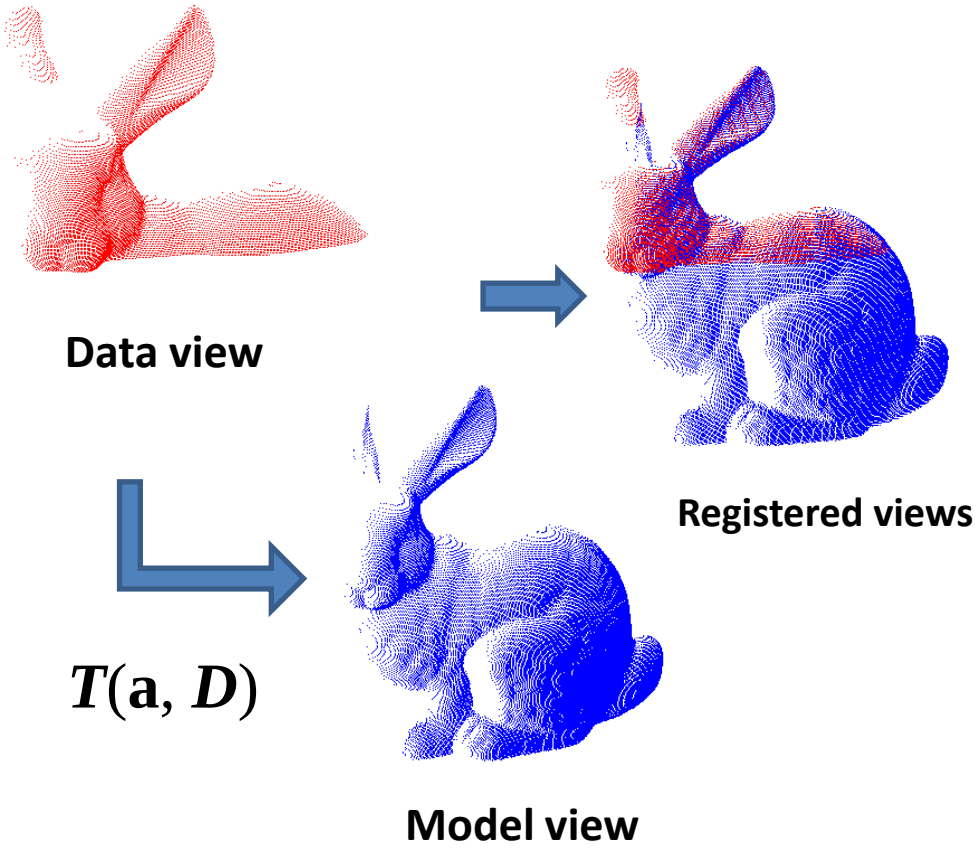
Two-view registration

Problem statement

$\boldsymbol{D}$ is the data view (moving)

$\boldsymbol{M}$ is the model view (fixed)

When the transformation T is applied to the data view, data view is moved to the model view and the alignment is obtained



Two-view registration

- **The transformation function**. The transformation function $\boldsymbol{T}$ usually implements a rigid transformation of the 3D space. It uses a translation vector $\mathbf{t}$ and a rotation matrix $\mathbf{R}$ whose values are encoded or parametrized in the parameter vector $\mathbf{a}$.

$$T(\mathbf{a}; \mathbf{D}) = \mathbf{RD} + \mathbf{t}$$

Transformation function for rigid registration

Note that the transformation function may also handle deformations but this requires a more complex formulation…

Two-view registration

- **The error function**. The error function $\boldsymbol{E}$ measures the registration error or dissimilarity between $\mathbf{D}$ and $\mathbf{M}$ after alignment. When the transformation function ***T*** is rigid, ***E*** is a measure of congruence between the two views.

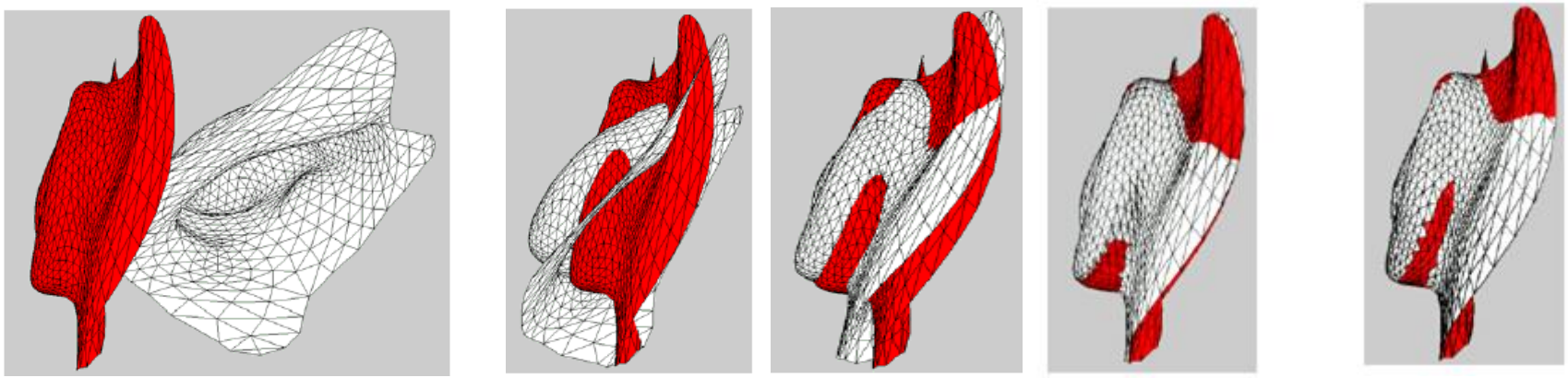
$$E = ||T(\mathbf{a};\mathbf{D}) - M||^2$$
$$= ||(\mathbf{RD} + \mathbf{t}) - M||^2$$

Error function for rigid registration (i.e., extrinsic similarity

Note that the error function may also handle deformations but this requires a more complex formulation…

Two-view registration

- **The optimisation method**. This is the method or algorithm used to find the minimizer of error function. The gold standard is the **ICP algorithm** which was specifically designed for the problem at hand.



Starting views **Iter 1** **Iter 2** **Iter 3** **…** **Iter N**

Note: General purpose optimisation methods such as **Levenberg-Marquardt** have also been used for this problem.

Rigid alignment

- When a set of corresponding points between $\mathbf{D}$ and $\mathbf{M}$ is available the rigid alignment can be estimated solving an **Orthogonal Procustes problem**
  - Data is represented by a set of points $\mathbf{d}_1, \ldots, \mathbf{d}_N$
  - Model is represented by a set of points $\mathbf{m}_1, \ldots., \mathbf{m}_N$

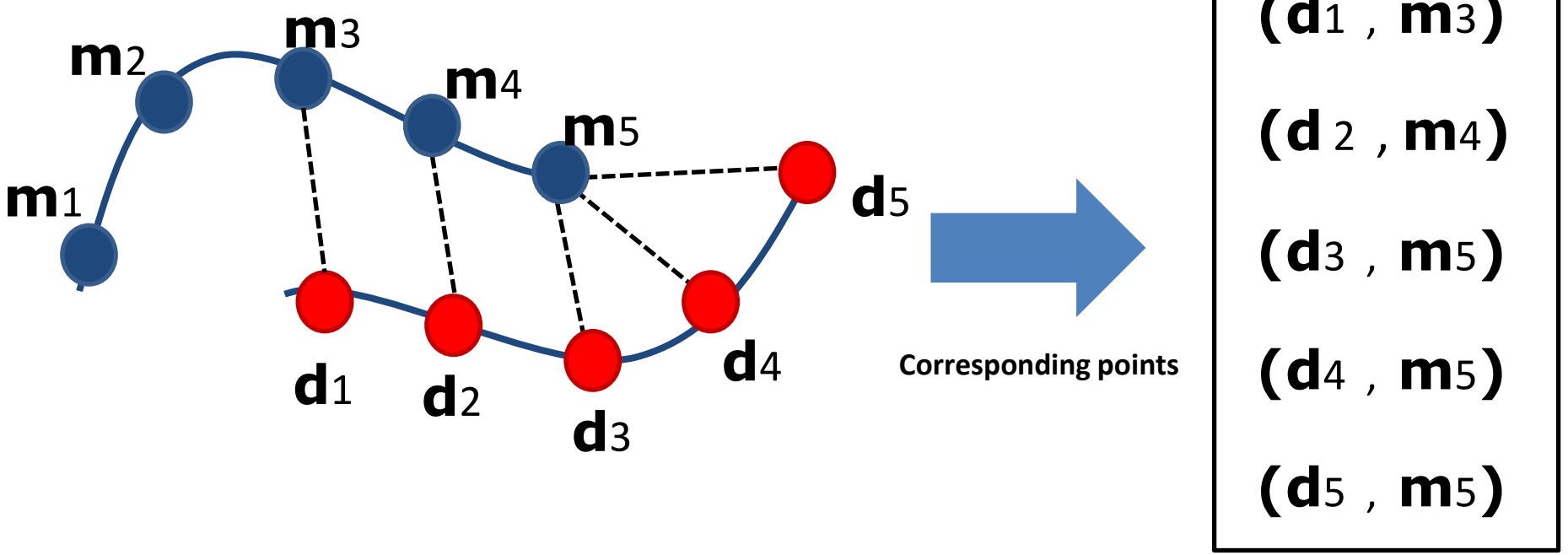
  Where each pair $(\mathbf{d}_i, \mathbf{m}_i)$ represent the same surface point observed from two different views

Computer vision course!

Iterative closest point (ICP)

- In standard scenarios the pairs ($\mathbf{d}_i$, $\mathbf{m}_i$) of corresponding points are not available!

Main idea: estimate the corresponding points with **closest points**.



Iterative closest point (ICP)

- Data $\mathbf{D}$ is represented by a set of points $\mathbf{d}_1$, …, $\mathbf{d}_{Nd}$
- Model $\mathbf{M}$ is represented by a set of points $\mathbf{m}_1$, …., $\mathbf{m}_{Nm}$
  - Fixing $\mathbf{d}_i \in \mathbf{D}$ the closest point $\mathbf{m}_j \in \mathbf{M}$ is computed such that:

$$j = \underset{j \in \{1,\ldots,N_m\}}{\arg\min} \left\| (\mathbf{R}\mathbf{d}_i + \mathbf{t}) - \mathbf{m}_j \right\|^2$$

Iterative closest point (ICP)

- The iterative closest point algorithm (ICP) for the registration of 3D points is based on the iteration between two main steps:
  - Estimate the pair of corresponding points ($\mathbf{d}_i$, $\mathbf{m}_j$) by computing the closest points,
  - Solve the Ortogonal Procustes problem to estimate $\mathbf{R}$ and $\mathbf{t}$

$$T(\mathbf{a};\mathbf{D}) = \mathbf{R}\mathbf{D} + \mathbf{t} \qquad \mathbf{a} = (\mathbf{R},\mathbf{t})$$ **Transformation function**

$$E_{ICP}(\mathbf{a},D,M) = \sum_{i=1}^{N_d} \|(\mathbf{R}\mathbf{d}_i + \mathbf{t}) - \mathbf{m}_j\|^2$$ **Error function**

Optimization method

Iterative closest point (ICP)

- **Summary of ICP algoritm:**

1. For each data-point $\mathbf{d}_i \in D$, compute the closest point $\mathbf{m}_j \in M$;

2. With the correspondences $(\mathbf{d}_i, \mathbf{m}_j)$ from step 1, compute the new transformation parameters $\mathbf{a} = (\mathbf{R}, \mathbf{t})$;
3. Apply the new transformation parameters $\mathbf{a}$ from step 2 to the point cloud D;
4. If the change in $E_{ICP}(\mathbf{a}, D, M)$ between two successive iterations is lower than a threshold terminate, else goto step 1.

This algorithm is guaranteed to converge monotonically to a local solution !

Iterative closest point (ICP)

- **Open issue** of standard ICP:
  - The two views must be close to each other. If not, ICP will probably get stuck in a local minimum.

a good initialization of **a** is needed!

Iterative closest point (ICP)

- **Open issue** of standard ICP:
  - The two views must fully overlap, or the data-view D must be a subset of the model-view M. Otherwise if a data point has no corresponding model point, this will create a spurious correspondence,

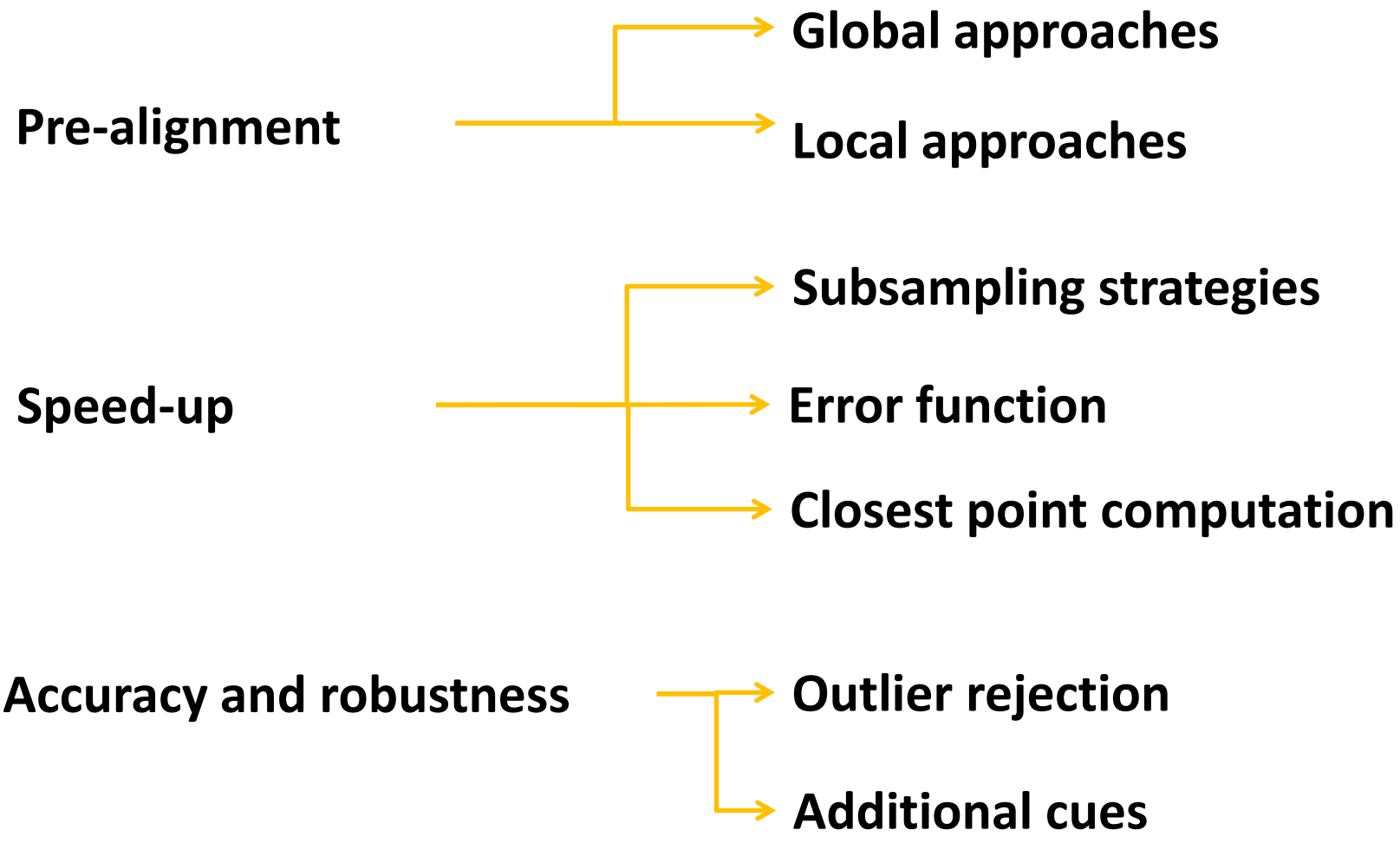
an outlier rejection strategy is needed!

Iterative closest point (ICP)

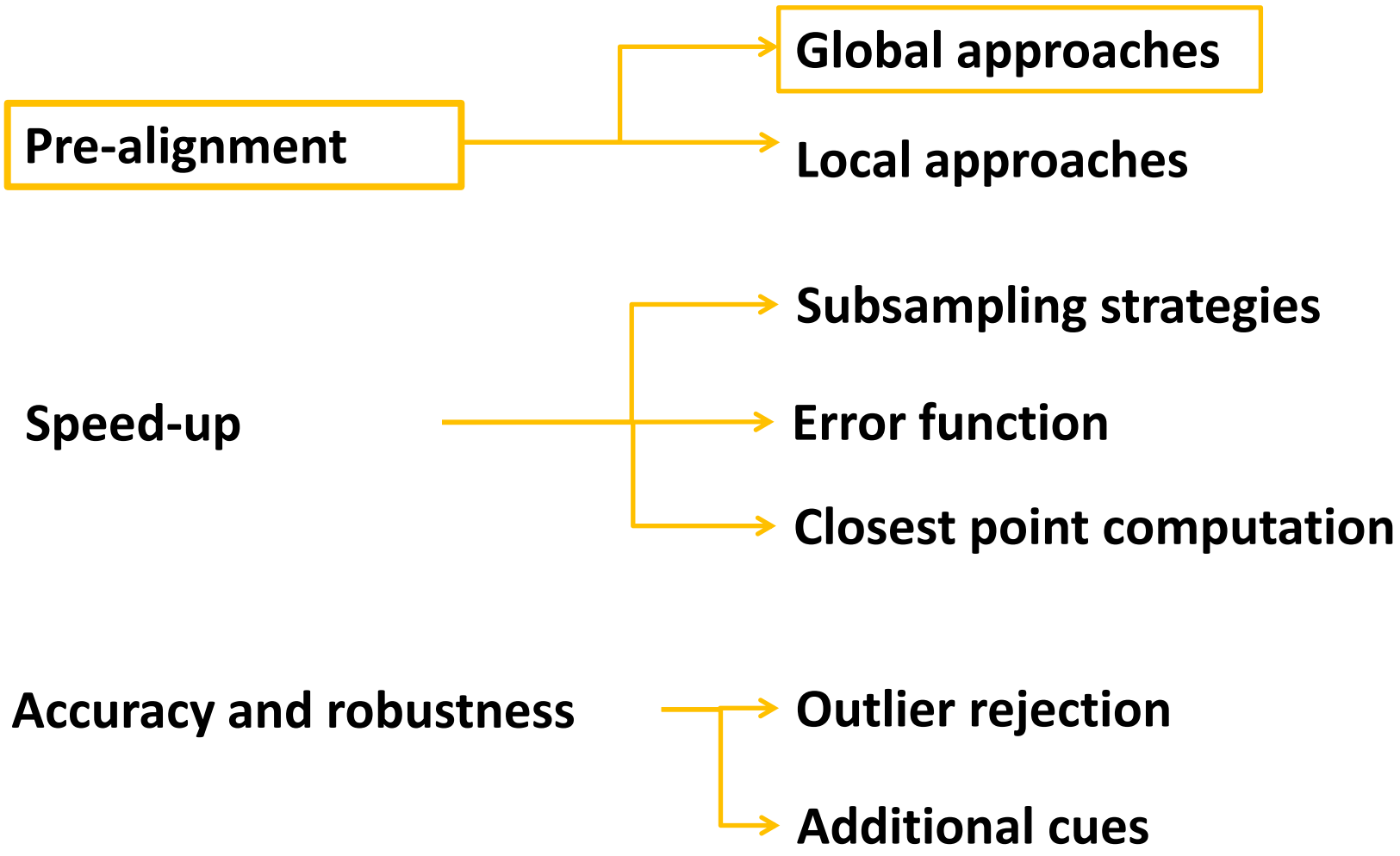
- **Open issue** of standard ICP:
  - The estimation of closest point is computational expensive,

A strategy to improve the speed is needed!

ICP variants



ICP variants



Pre-alignment

Global pre-alignment (i.e., all points are used)

$$x' = Rx + t$$

- How to remove **translation** and **rotation** ambiguity?
- Find some "**canonical**" placement of the shape X in $\mathbb{R}^3$

Pre-alignment

- **Extrinsic centroid** (a.k.a. **center of mass**, or **center of gravity**):

$$x_0 = \int_X x dx$$

- Set $t = -x_0$ to resolve translation ambiguity.
- Three degrees of freedom remaining…

Pre-alignment

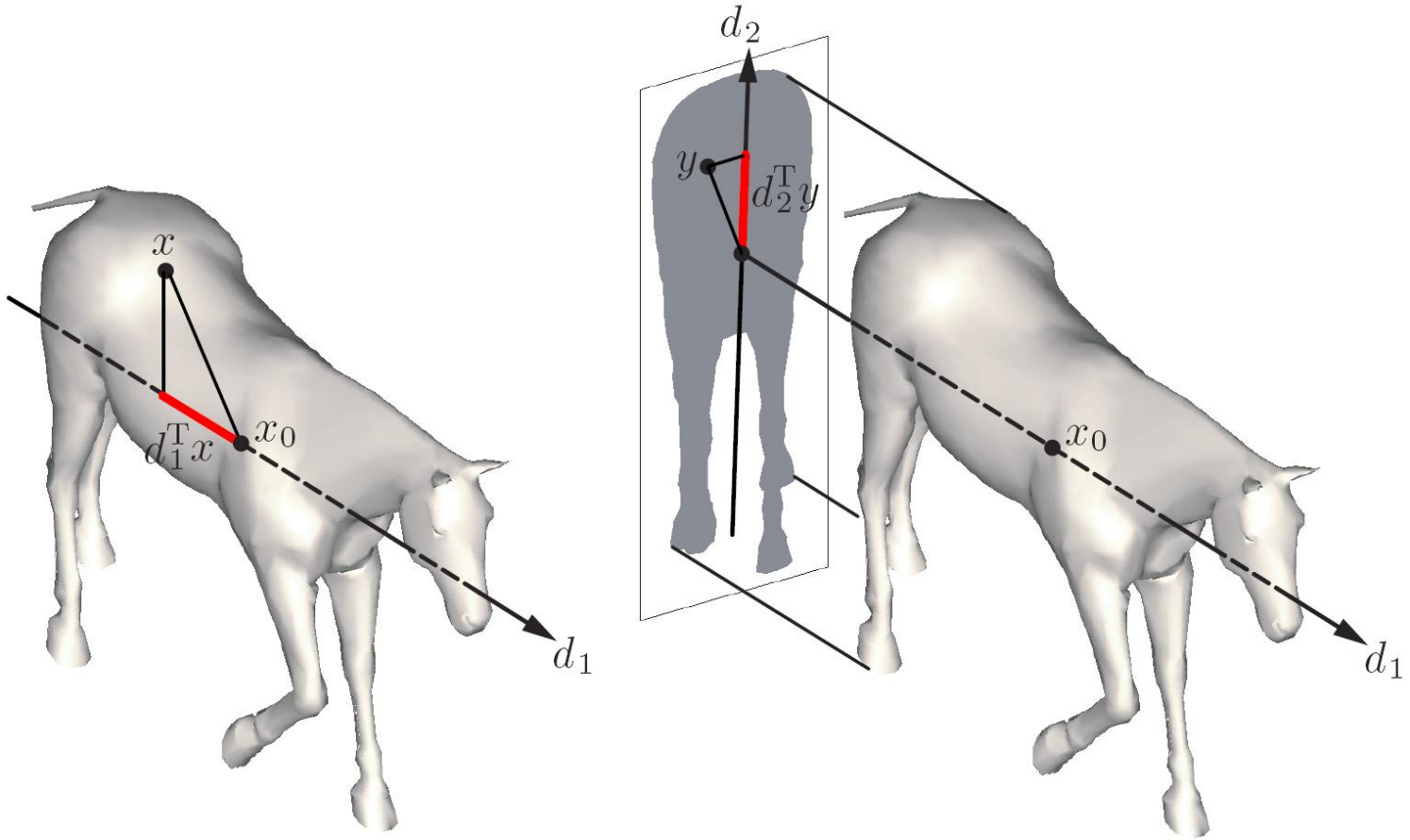
- Find the direction d_1 in which the surface has **maximum extent**.
- Maximize **variance** of projection of X onto d_1

$$\begin{aligned} d_1 &= \arg \max_{d_1 : \|d_1\|_2 = 1} \int_X (d^\top x)^2 dx \\ &= \arg \max_{d_1 : \|d_1\|_2 = 1} d_1^\top \left(\int_X x x^\top dx \right) d_1 \\ &= \arg \max_{d_1 : \|d_1\|_2 = 1} d_1^\top \Sigma_X d_1 \end{aligned}$$

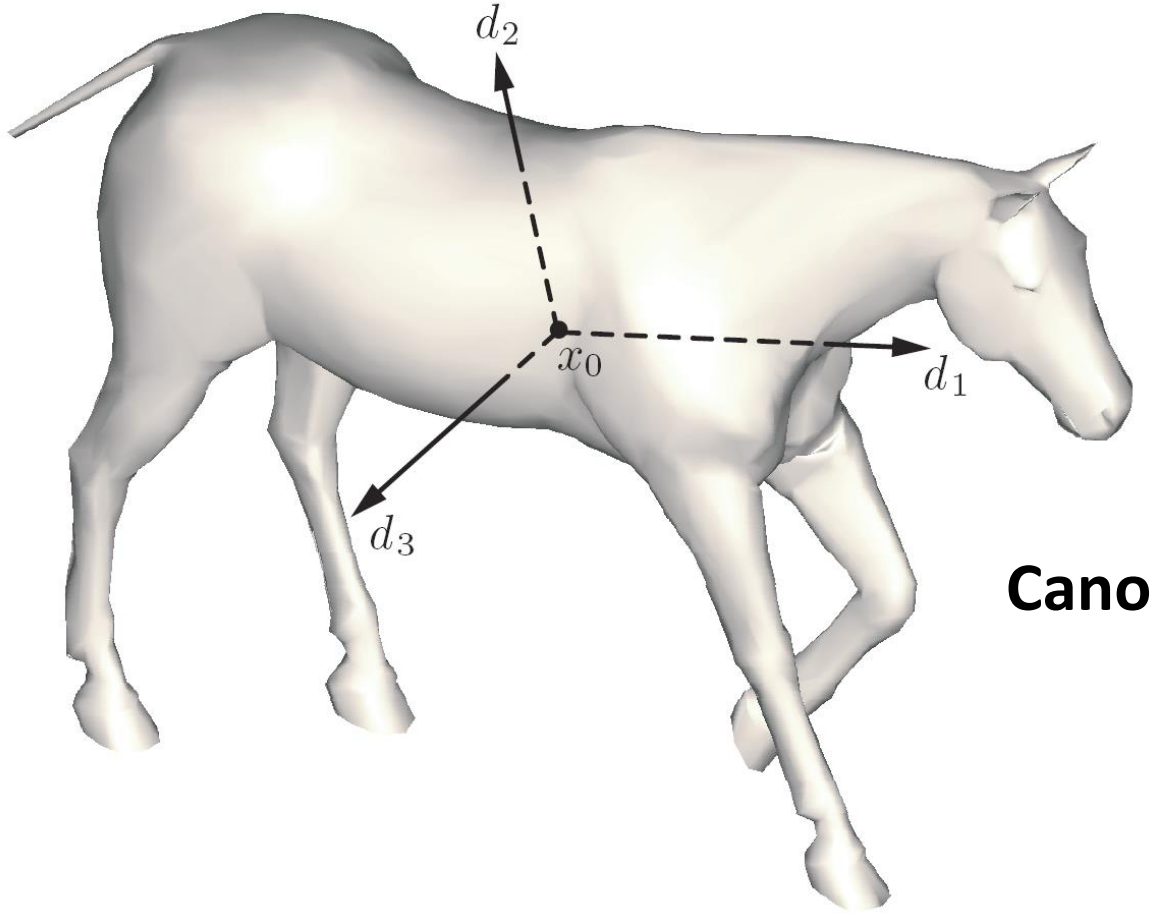
- Σ_X is the **covariance matrix**
- **Second-order geometric moments** of X: $\sigma_{ij} = \int_X x^i x^j dx$
- d_1 is the first **principal direction**

Pre-alignment

- Project X on the plane orthogonal to d_1.
- Repeat the process to find second and third principal directions d_2, d_3.



Pre-alignment



Canonical basis

- $d_1 \perp d_2 \perp d_3$ span a canonical orthogonal basis for X in $\mathbb{R}^3$.

How to get rid of the rotation ambiguity?

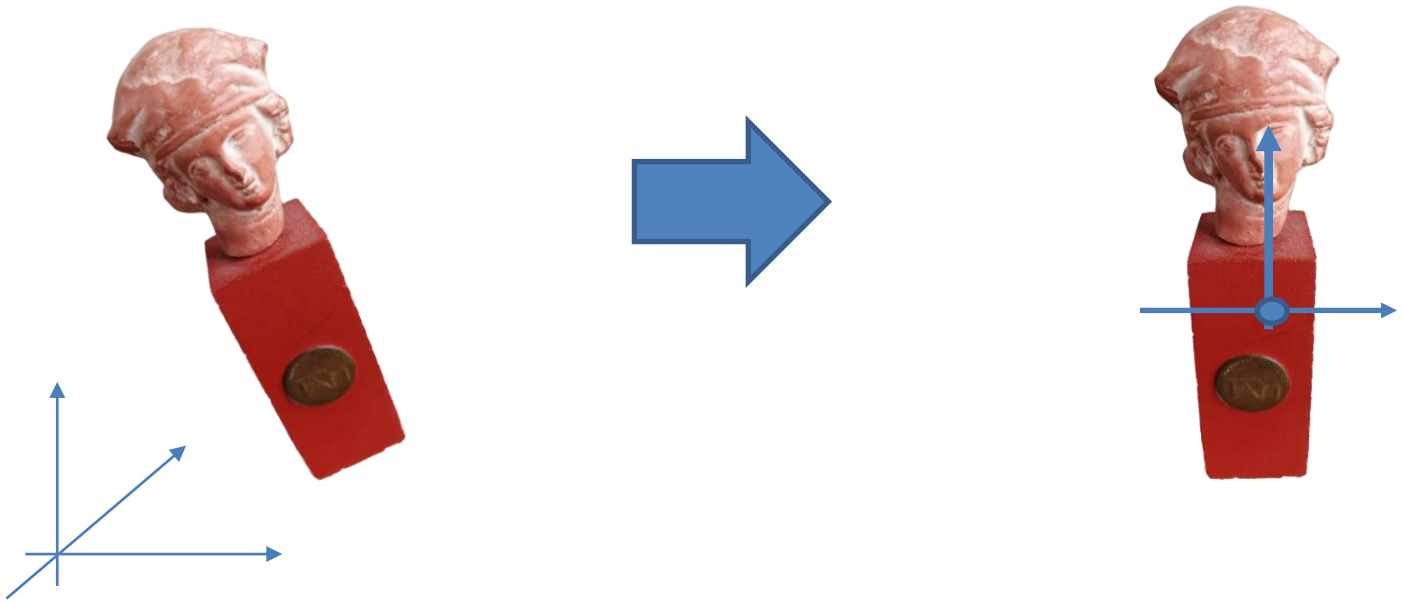
- Direction maximizing $d_1^\top \Sigma_X d_1$ = **largest eigenvector** of Σ_X.
- d_2 and d_3 correspond to the second and third eigenvectors of Σ_X.
- Σ_X admits **unitary diagonalization** $\Sigma_X = U^\top \Lambda U$.
- Setting $R = U^\top$ aligns d_1, d_2, d_3 with the **standard basis** axes e_1, e_2, e_3.
- **Principal component analysis** (PCA), a.k.a. **Karhunen-Loéve transform** (KLT), or **Hotelling transform**.
- Bottom line: the transformation

$$(R, t) \;=\; (U^\top, -U^\top x_0)$$

brings the shape into a canonical configuration in $\mathbb{R}^3$.

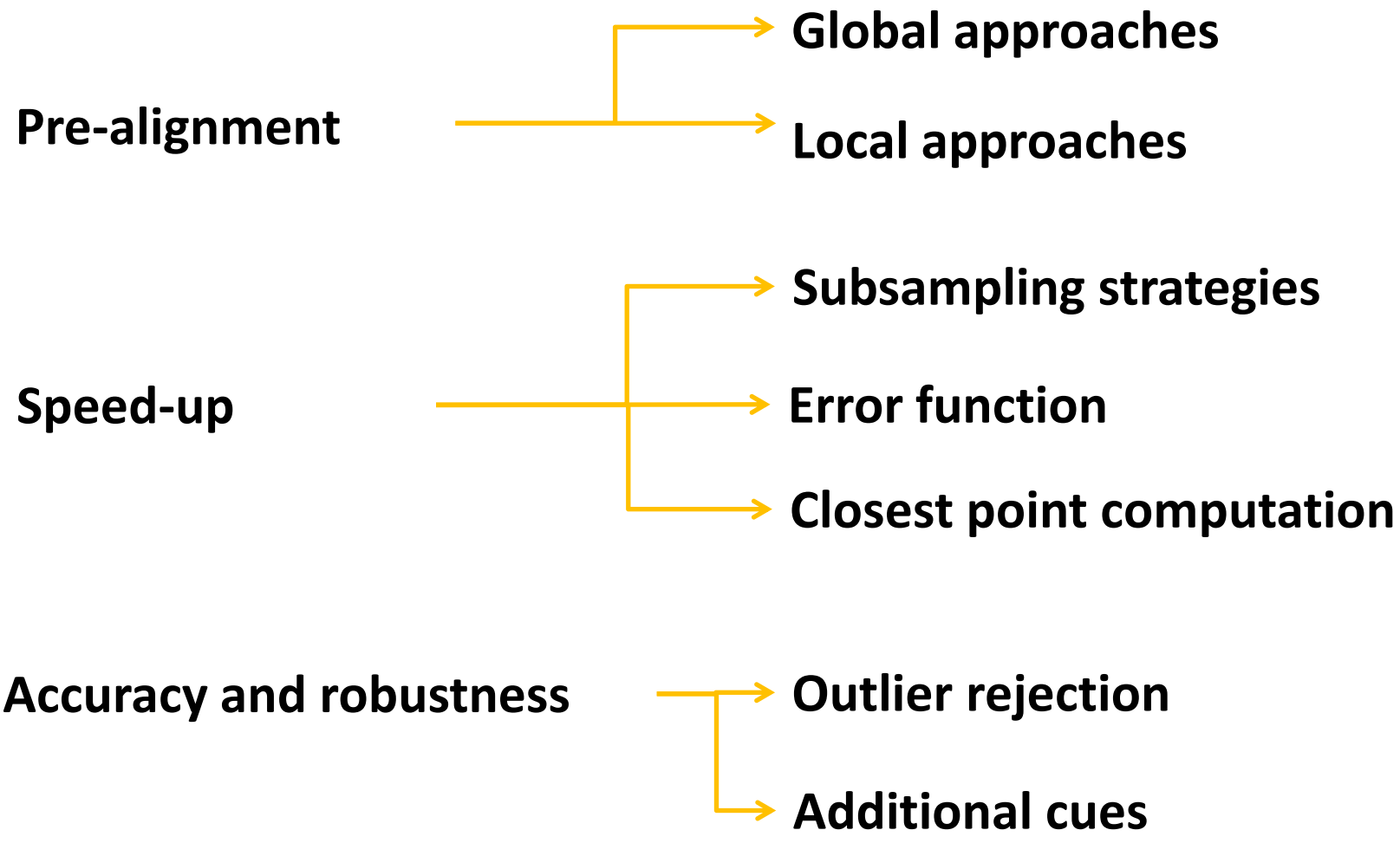
Homework 3

Compute the canonical orientation of your object:



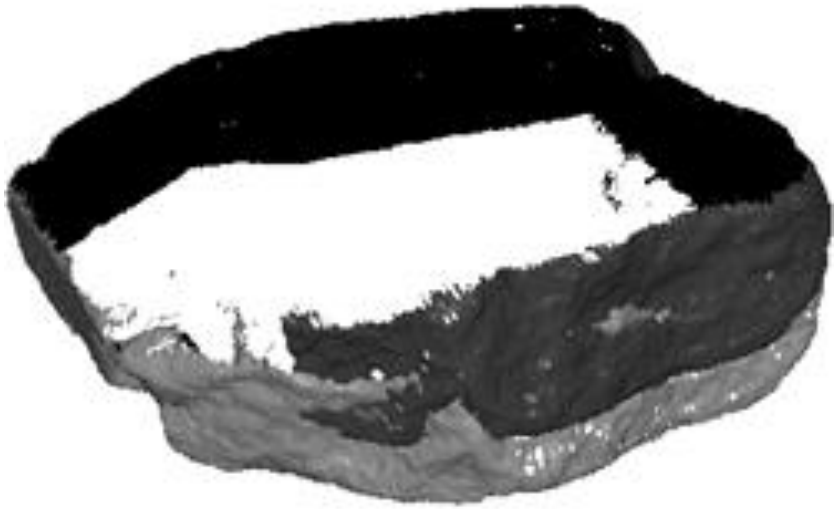
Zephyr brings the object to an arbitrary position

ICP variants



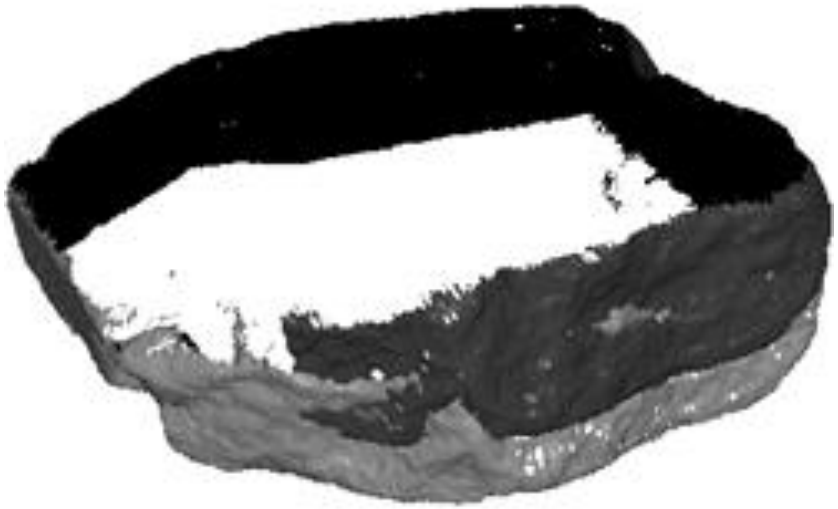
Global Registration Goal

- Given: *n* scans around an object
- Goal: align them all
- First attempt: ICP each scan to one other and then bring views to global ref system by concatenation



Global Registration Goal

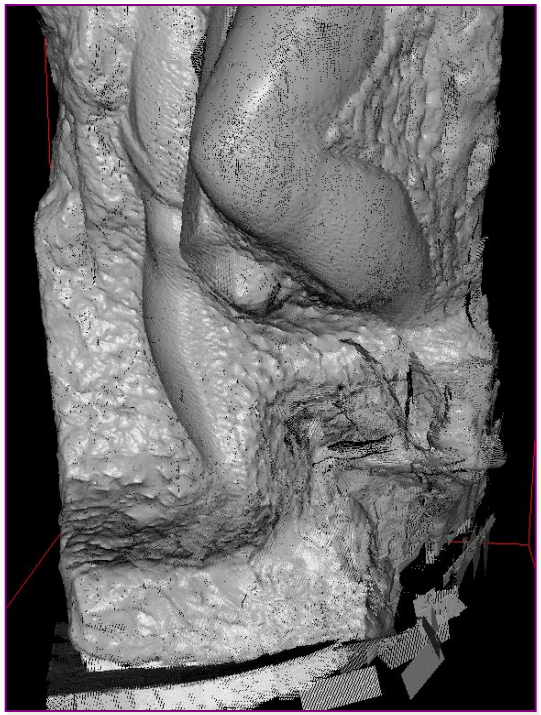
- Problem: small errors from pairwise views are amplified when sequential views are involved
- Advanced method for distributing accumulated error among all scans



Bad ICP in Globalreg

One bad ICP can throw off the entire model!

Correct Globalreg



Globalreg Including Bad ICP

Matlab excercise

- Globa alignment by accumulation of rigid transforms