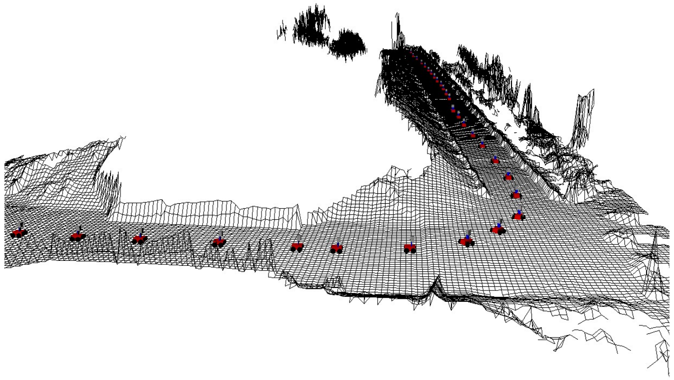
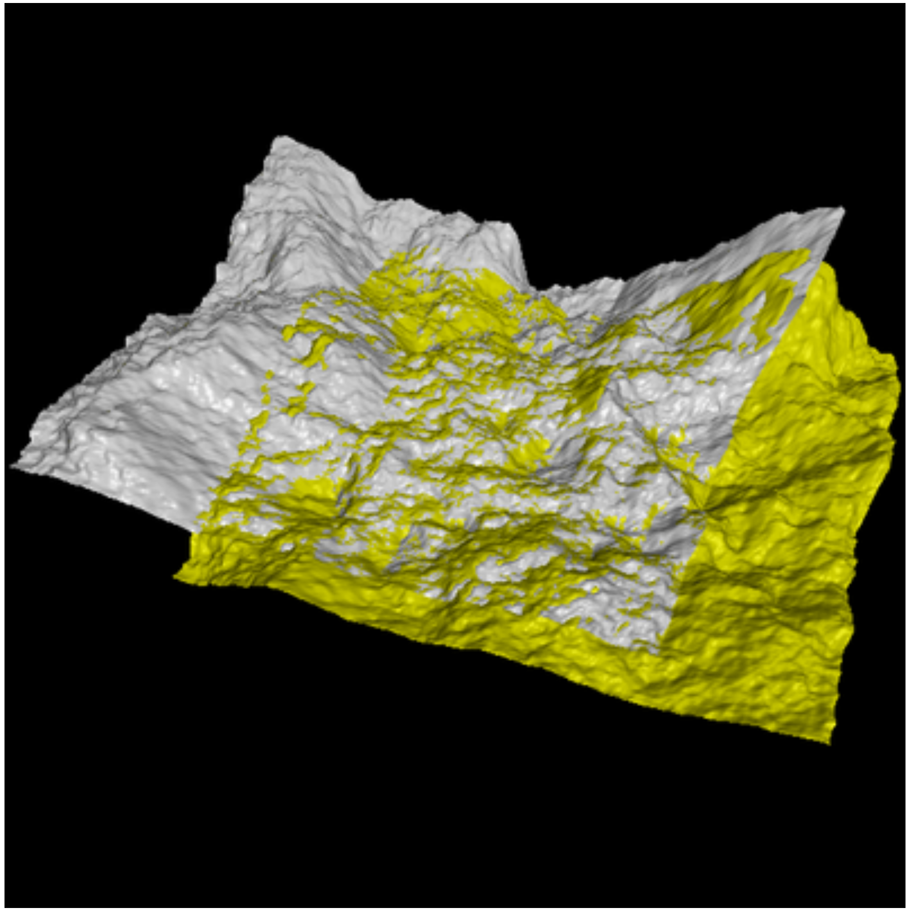
Photogrammetry & Robotics Lab

Point Cloud Registration & ICP #1: Known Data Association

Cyrill Stachniss

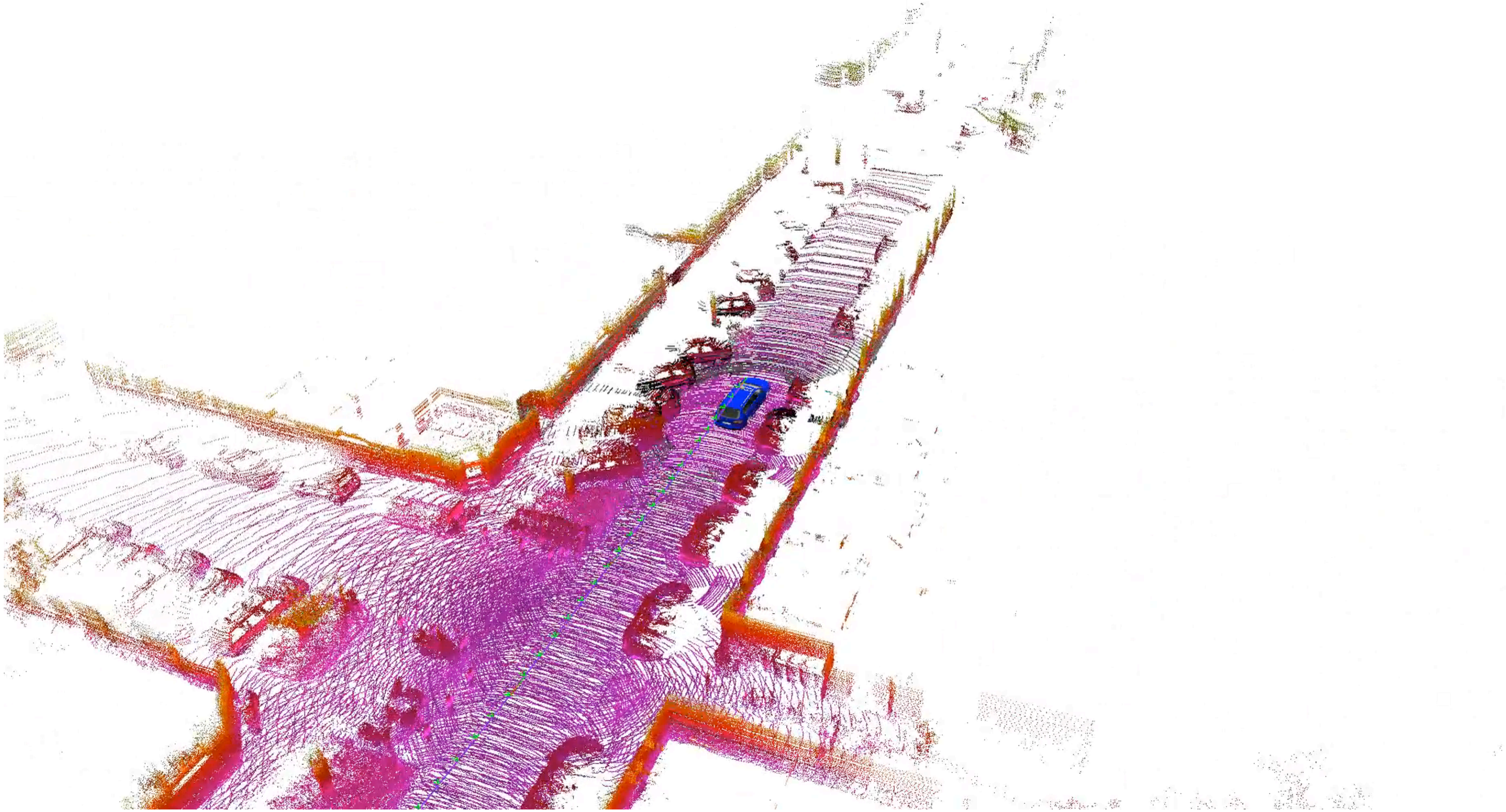
Scan Alignment in Mapping



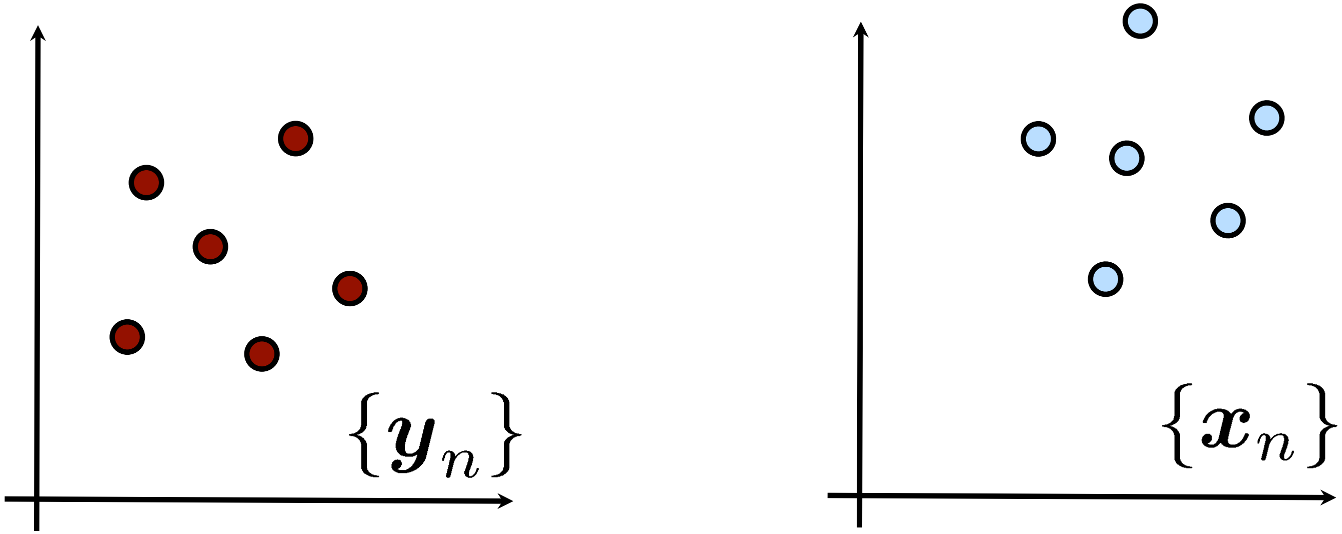


Goal: Find local transformation to align points

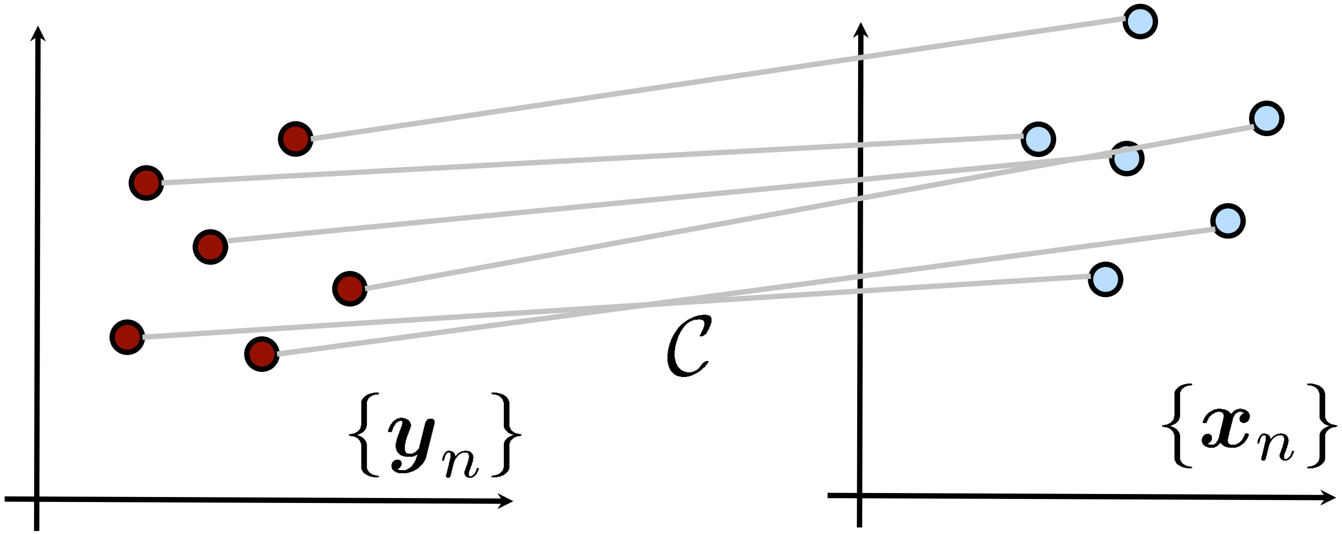
3D Scan Registration



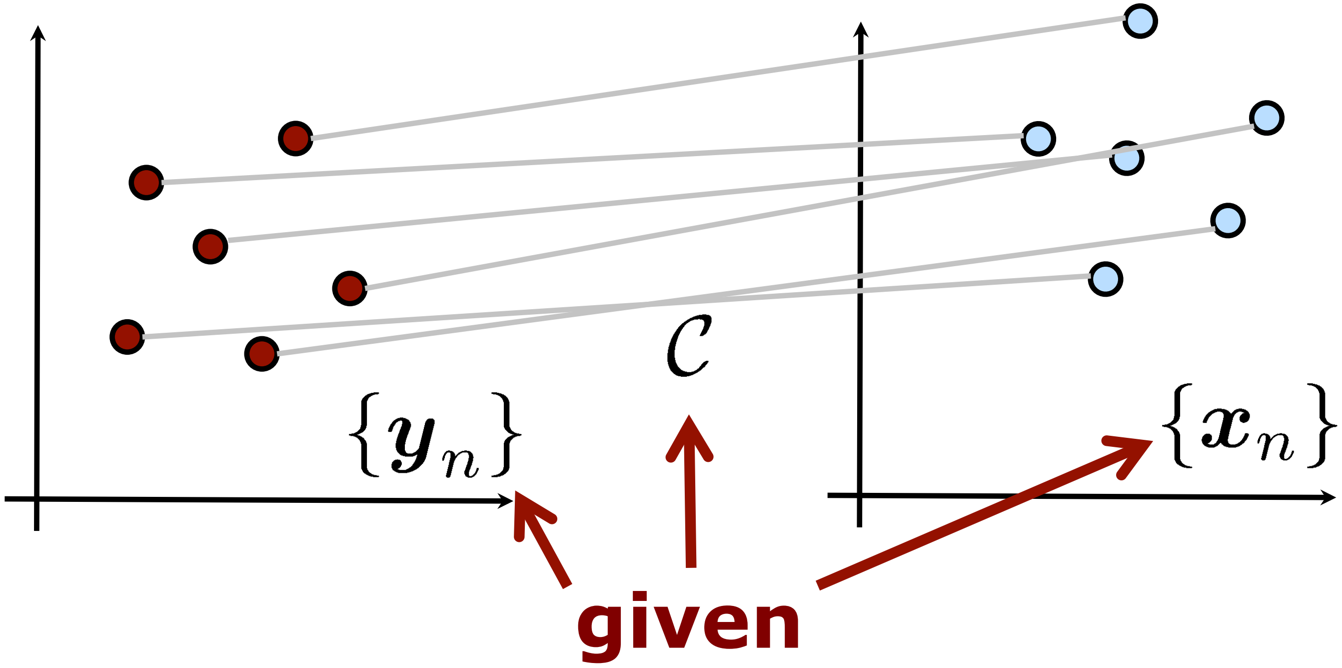
Simple Form of Point Cloud Registration



Simple Form of Point Cloud Registration

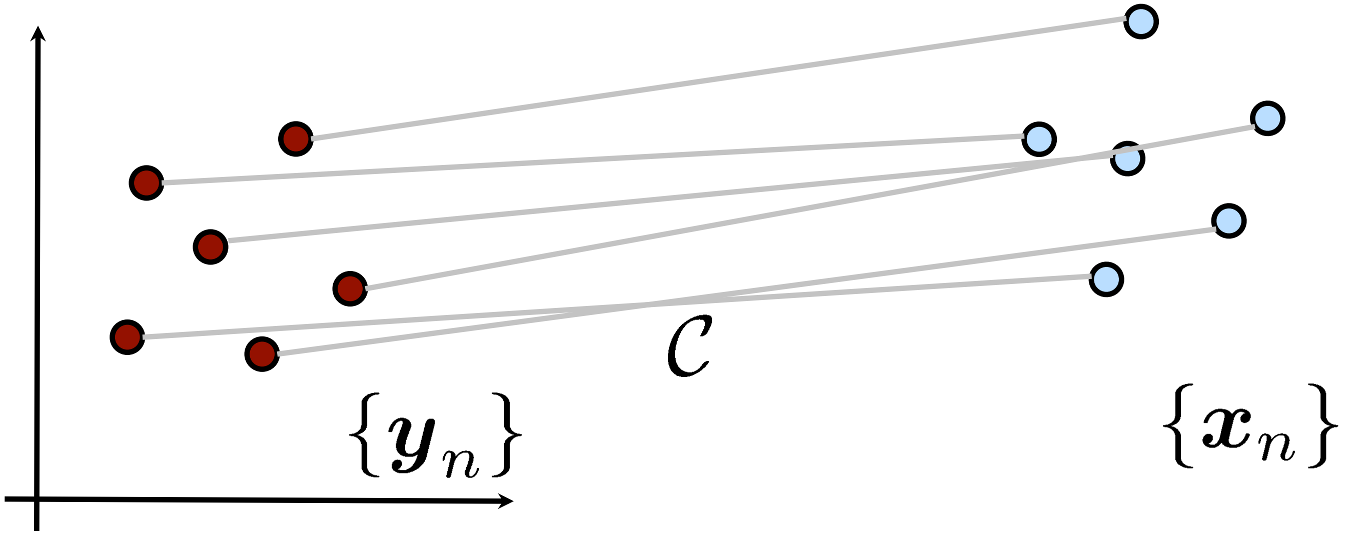


Simple Form of Point Cloud Registration

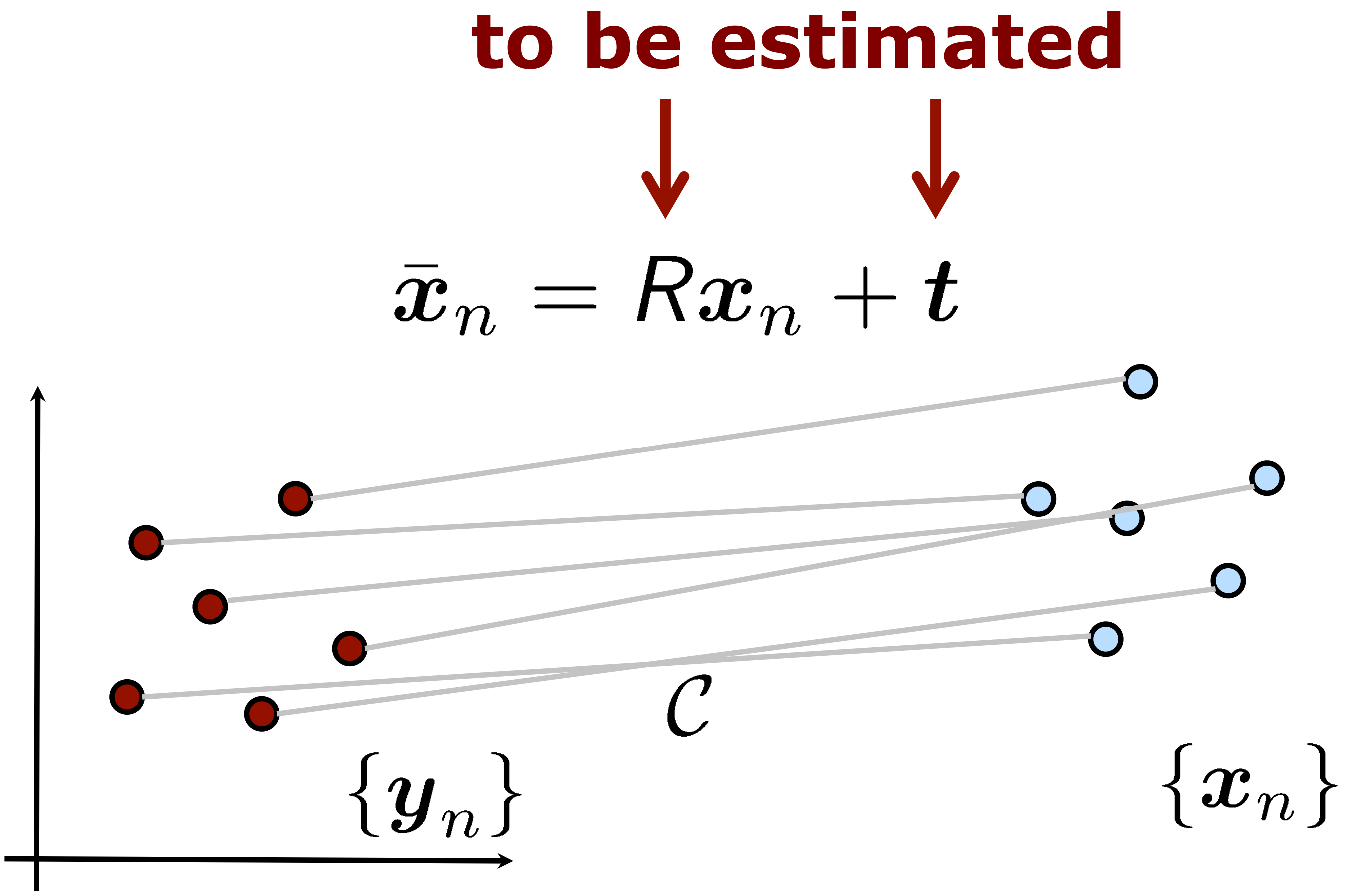


Simple Form of Point Cloud Registration

$$\bar{\boldsymbol{x}}_n = R\boldsymbol{x}_n + \boldsymbol{t}$$



Simple Form of Point Cloud Registration

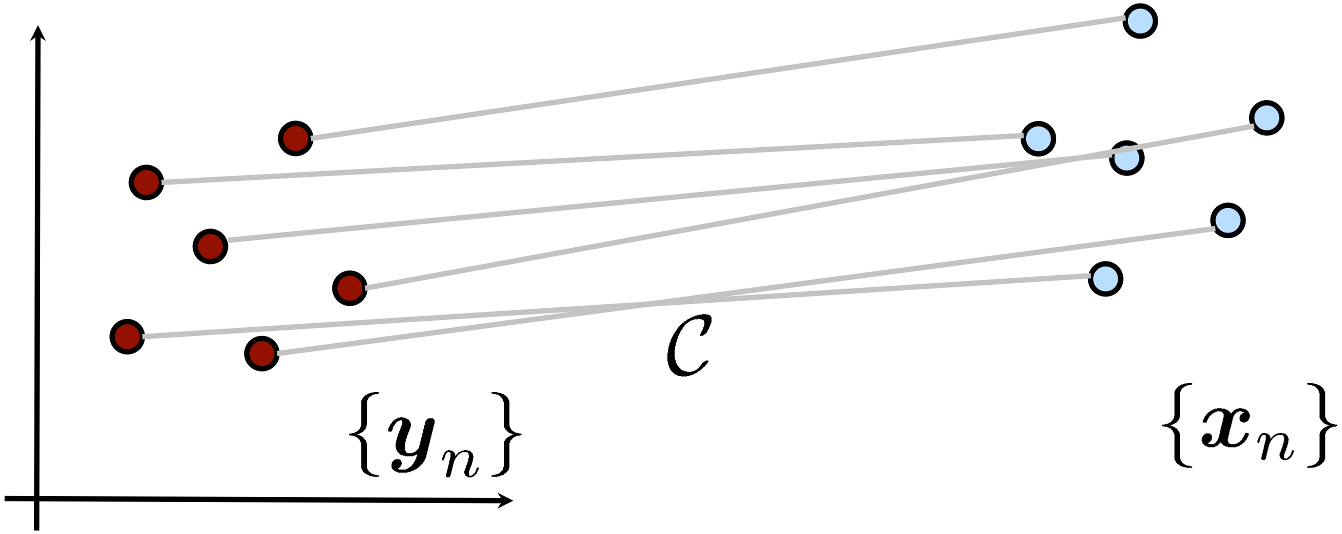


Simple Form of Point Cloud Registration

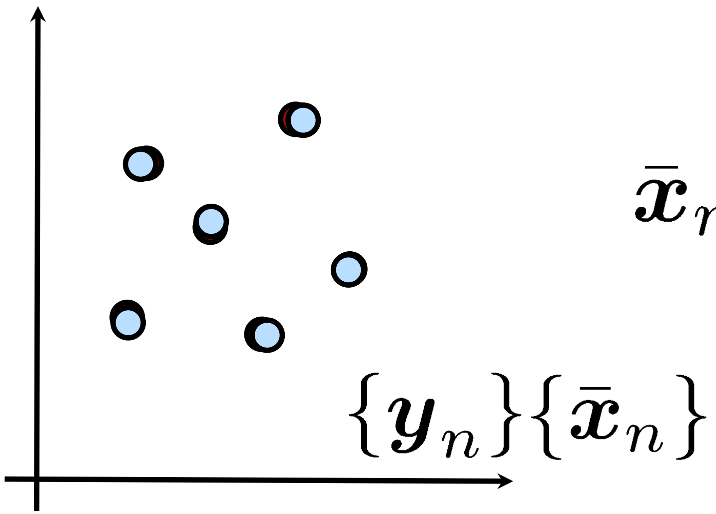
transformation should yield

$$\bar{\boldsymbol{x}}_n = R\boldsymbol{x}_n + \boldsymbol{t}$$

$$\sum ||\boldsymbol{y}_n - \bar{\boldsymbol{x}}_n||^2 \rightarrow \min$$



Simple Form of Point Cloud Registration



$$\bar{\boldsymbol{x}}_n = R\boldsymbol{x}_n + \boldsymbol{t}$$

Registration of 3D Data Points

- **Goal:** find the parameters of the transformation that best align corresponding data points
- Optimization / search for parameters
  - Iterative closest point (ICP w/ SVD)
  - Robust least squares approaches (#3)
- **Known (#1)** vs. estimated (#2) correspondences

3 parts

Part 1
Point Cloud Registration with Known Data Association

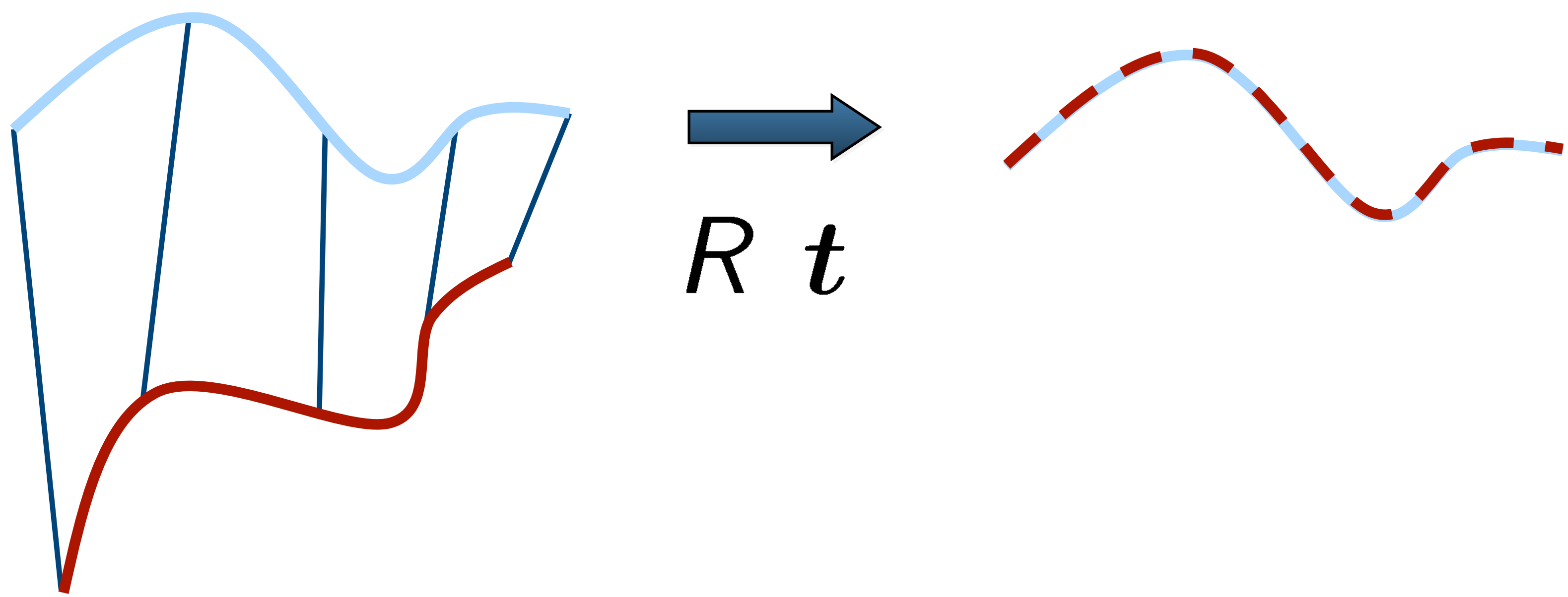
The Basic Alignment Problem

- Given two input point sets:
  $Y = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_I\} \qquad X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_J\}$
  with correspondences $\mathcal{C} = \{(i,j)\}$
- Wanted: Translation $\boldsymbol{t}$ and rotation *R* that minimize the sum of the squared errors:

$$\sum_{(i,j)\in\mathcal{C}} \|\boldsymbol{y}_i - R\boldsymbol{x}_j - \boldsymbol{t}\|^2 \to \min$$

Key Idea

Given correct correspondences compute a shift and rotation to align the points using a direct solution.



Simplified Correspondences

- Reorder point clouds X, Y given the correspondences C using an index n
- Point sets: $\{\boldsymbol{x}_n\}\{\boldsymbol{y}_n\}$
- Find the rigid body transform

$$\bar{\boldsymbol{x}}_n = \boldsymbol{R}\boldsymbol{x}_n + \boldsymbol{t} \qquad n = 1, \ldots, |C| =: N$$

- That transforms the points $\{\boldsymbol{x}_n\}$ into $\{\bar{\boldsymbol{x}}_n\}$
- So that the point set $\{\boldsymbol{x}_n\}$ will be as close as possible to the point set $\{\boldsymbol{y}_n\}$
- Minimizing the sum of squared point-to-point distances

Special Case of the Absolute Orientation Problem

- In the absolute orientation problem, we look for the similarity transform

$$\bar{\boldsymbol{x}}_n = \lambda R \boldsymbol{x}_n + \boldsymbol{t}$$

- transforming 3D point sets
- Here, we only need the rigid body transform, i.e., $\lambda = 1$, so that

$$\bar{\boldsymbol{x}}_n = R \boldsymbol{x}_n + \boldsymbol{t}$$

Formal Problem Definition

- Given corresponding points:

$$\boldsymbol{y}_n, \boldsymbol{x}_n \qquad n = 1, \ldots, N$$

- and optionally weights:

$$p_n \qquad n = 1, \ldots, N$$

- Find the parameters $R, \boldsymbol{t}$ of the rigid body transform with

$$\bar{\boldsymbol{x}}_n = R\boldsymbol{x}_n + \boldsymbol{t} \qquad n = 1, \ldots, N$$

- so that the squared error is minimized

$$\sum ||\boldsymbol{y}_n - \bar{\boldsymbol{x}}_n||^2 p_n \rightarrow \min$$

Direct Optimal Solution Exists

- There exists a direct and optimal solution solving $\sum ||\boldsymbol{y}_n - \bar{\boldsymbol{x}}_n||^2 \, p_n \to \min$
- **Direct** = no initial guess needed
- **Optimal** = no better solution exists

Informally speaking:

- Computes a **shift** involving the **center of masses** of both point clouds
- Performs a **rotational** alignment using singular value decomposition (**SVD**)

Direct Computing of the Rotation Matrix

$$\boldsymbol{x}_0 = \frac{\sum \boldsymbol{x}_n p_n}{\sum p_n} \qquad \boldsymbol{y}_0 = \frac{\sum \boldsymbol{y}_n p_n}{\sum p_n}$$

$$H = \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)(\boldsymbol{y}_n - \boldsymbol{y}_0)^\top p_n$$

$$\mathrm{svd}(H) = UDV^\top$$

$$R = VU^\top$$

Direct Computing of the Translation Vector

$$\boldsymbol{y}_0 = \frac{\sum \boldsymbol{y}_n p_n}{\sum p_n} \qquad R = VU^\top \qquad \boldsymbol{x}_0 = \frac{\sum \boldsymbol{x}_n p_n}{\sum p_n}$$

$$\boldsymbol{t} = \boldsymbol{y}_0 - R\boldsymbol{x}_0$$

Solution for Computing the Rigid Body Transform

- Rotation
- Translation

$$\boxed{\begin{aligned} R &= VU^\top \\ \boldsymbol{t} &= \boldsymbol{y}_0 - R\boldsymbol{x}_0 \end{aligned}}$$

- with

$$H = \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)(\boldsymbol{y}_n - \boldsymbol{y}_0)^\top p_n \quad \mathrm{svd}(H) = UDV^\top$$

$$\boldsymbol{y}_0 = \frac{\sum \boldsymbol{y}_n p_n}{\sum p_n} \qquad \boldsymbol{x}_0 = \frac{\sum \boldsymbol{x}_n p_n}{\sum p_n}$$

SVD-Based Alignment (1)

- Compute means of the point sets

$$\boldsymbol{y}_0 = \frac{\sum \boldsymbol{y}_n p_n}{\sum p_n} \qquad \boldsymbol{x}_0 = \frac{\sum \boldsymbol{x}_n p_n}{\sum p_n}$$

- Compute cross covariance matrix based on mean-reduced coordinates

$$H = \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)(\boldsymbol{y}_n - \boldsymbol{y}_0)^\top p_n$$

SVD-Based Alignment (2)

- Compute SVD

$$\mathrm{svd}(H) = UDV^\top$$

- Rotation matrix is given by

$$R = UV^\top$$

- Translation vector is given by:

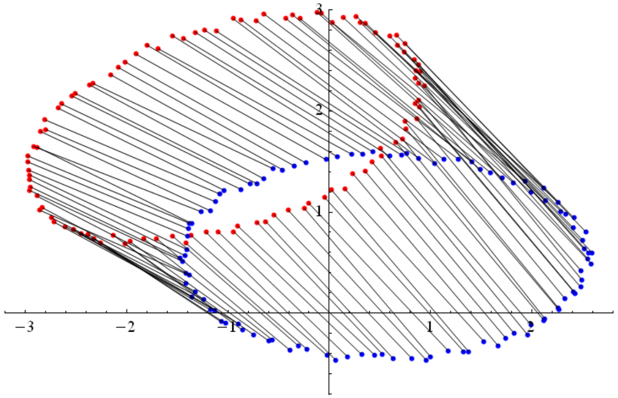
$$\boldsymbol{t} = \boldsymbol{y}_0 - R\boldsymbol{x}_0$$

- Translate and rotate points:

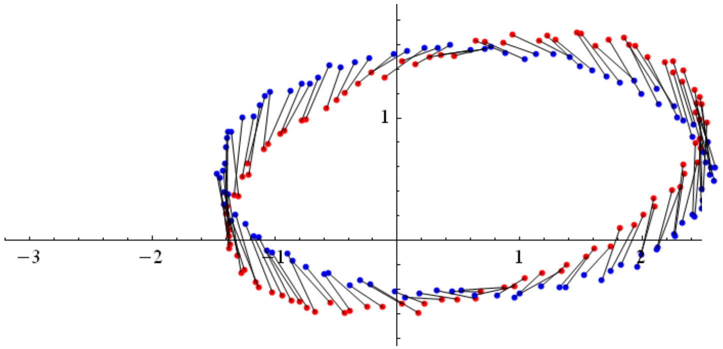
$$\bar{\boldsymbol{x}}_n = R\boldsymbol{x}_n + \boldsymbol{t} \qquad n = 1, \ldots, N$$

SVD-Based Alignment Summary

Alignment through translation and rotation $\bar{\boldsymbol{x}}_n = R(\boldsymbol{x}_n - \boldsymbol{x}_0) + \boldsymbol{y}_0$



translate points to make the center of masses overlap



rotate points

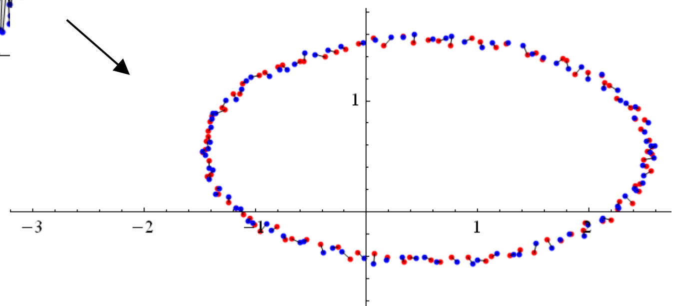


Image courtesy: Ju

We are done!

Why is this a Good Solution?

Let’s Start from the Beginning and Derive of the Solution

Arun et al. “Least-Squares Fitting of Two 3D Point Sets”, IEEE T-PAMI 9(5), 698–700, 1987.

Formal Problem Definition

- Given corresponding points and weights $\boldsymbol{y}_n, \boldsymbol{x}_n, p_n \qquad n = 1, \ldots, N$

- Find the parameters $R, \boldsymbol{t}$ of the rigid body transform

$$\bar{\boldsymbol{x}}_n = R\boldsymbol{x}_n + \boldsymbol{t} \qquad n = 1, \ldots, N$$

- So that the weighted sum the squared errors is minimized

$$\sum ||\boldsymbol{y}_n - \bar{\boldsymbol{x}}_n||^2 p_n \rightarrow \min$$

Use Local Coordinate System

- We want to use local coordinates defined by the point set $\{\boldsymbol{y}_n\}$
- We set the origin as weighted mean of $\{\boldsymbol{y}_n\}$ computed by

$$\boldsymbol{y}_0 = \frac{\sum \boldsymbol{y}_n p_n}{\sum p_n}$$

- so that we minimize

$$\sum ||\boldsymbol{y}_n - \boldsymbol{y}_0 - R\boldsymbol{x}_n - \boldsymbol{t} + \boldsymbol{y}_0||^2 \, p_n \to \min$$

does not change the problem

Rewrite Translation Vector

- Start with $\bar{\boldsymbol{x}}_n = R\boldsymbol{x}_n + \boldsymbol{t}$
- and use the shift of the origin

$$\bar{\boldsymbol{x}}_n - \boldsymbol{y}_0 = R\boldsymbol{x}_n + \boldsymbol{t} - \boldsymbol{y}_0$$

- to rewrite the translation vector

$$\bar{\boldsymbol{x}}_n - \boldsymbol{y}_0 = R(\boldsymbol{x}_n + \underline{R^\top \boldsymbol{t} - R^\top \boldsymbol{y}_0})$$

- Introduce a **new variable** $\boldsymbol{x}_0$:

$$\bar{\boldsymbol{x}}_n - \boldsymbol{y}_0 = R(\boldsymbol{x}_n - \boldsymbol{x}_0)$$

- with $\boldsymbol{x}_0 = R^\top \boldsymbol{y}_0 - R^\top \boldsymbol{t}$

Minimization Problem

- The initially formulated problem

$$\sum ||\boldsymbol{y}_n - \bar{\boldsymbol{x}}_n||^2 \, p_n \to \min$$

- turns into

$$\sum ||\boldsymbol{y}_n - \boldsymbol{y}_0 - R(\boldsymbol{x}_n - \boldsymbol{x}_0)||^2 \, p_n \to \min$$

- We need to find $R, \boldsymbol{x}_0$ so that

$$R^*, \boldsymbol{x}_0^* = \underset{R, \boldsymbol{x}_0}{\operatorname{argmin}} \sum ||\boldsymbol{y}_n - \boldsymbol{y}_0 - R(\boldsymbol{x}_n - \boldsymbol{x}_0)||^2 \, p_n$$

Define the Objective Function

- Minimize the function $\Phi(\boldsymbol{x}_0, R)$
- defined by

$$\Phi(\boldsymbol{x}_0, R) = \sum [(\boldsymbol{y}_n - \boldsymbol{y}_0) - R(\boldsymbol{x}_n - \boldsymbol{x}_0)]^\top \\ [(\boldsymbol{y}_n - \boldsymbol{y}_0) - R(\boldsymbol{x}_n - \boldsymbol{x}_0)]\, p_n$$

How to minimize this function?

Minimize Objective Function

- Minimize the objective function

$$
\begin{aligned}
\Phi(\boldsymbol{x}_0, R) \quad = \quad \sum & \left[(\boldsymbol{y}_n - \boldsymbol{y}_0) - R(\boldsymbol{x}_n - \boldsymbol{x}_0) \right]^\top \\
& \left[(\boldsymbol{y}_n - \boldsymbol{y}_0) - R(\boldsymbol{x}_n - \boldsymbol{x}_0) \right] p_n
\end{aligned}
$$

Solve $R^*, \boldsymbol{x}_0^* = \operatorname{argmin} \Phi(\boldsymbol{x}_0, R)$ **by**

- Computing the first derivatives
- Setting derivatives to zero
- Solving the resulting equations

Rearrange the Terms

- Rearrange the objective function

$$\begin{aligned}\Phi(\boldsymbol{x}_0, R) &= \sum \left[(\boldsymbol{y}_n - \boldsymbol{y}_0) - R(\boldsymbol{x}_n - \boldsymbol{x}_0)\right]^\top \\ &\quad \left[(\boldsymbol{y}_n - \boldsymbol{y}_0) - R(\boldsymbol{x}_n - \boldsymbol{x}_0)\right] p_n\end{aligned}$$

- to

$$\begin{aligned}\Phi(\boldsymbol{x}_0, R) &= \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\top (\boldsymbol{y}_n - \boldsymbol{y}_0)\, p_n \\ &\quad + \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)^\top (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n \\ &\quad - 2 \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\top R (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n\end{aligned}$$

Rearrange the Terms

- Rearrange the objective function

$$\Phi(\boldsymbol{x}_0, R) = \sum \left[(\boldsymbol{y}_n - \boldsymbol{y}_0) - R(\boldsymbol{x}_n - \boldsymbol{x}_0)\right]^\top \left[(\boldsymbol{y}_n - \boldsymbol{y}_0) - R(\boldsymbol{x}_n - \boldsymbol{x}_0)\right] p_n$$

- to

$$\begin{aligned}
\Phi(\boldsymbol{x}_0, R) = & \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\top (\boldsymbol{y}_n - \boldsymbol{y}_0)\, p_n \longleftarrow \text{no } \boldsymbol{x}_0, R \\
& + \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)^\top (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n \longleftarrow \text{no } R \\
& - 2 \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\top R (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n
\end{aligned}$$

Solve w.r.t. $\boldsymbol{x}_0$

Derivative with respect to $\boldsymbol{x}_0$

- Compute first derivative of

$$\begin{aligned}\Phi(\boldsymbol{x}_0, R) \quad &= \quad \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\top (\boldsymbol{y}_n - \boldsymbol{y}_0)\, p_n \\ &\quad + \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)^\top (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n \\ &\quad -2 \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\top R (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n\end{aligned}$$

- with respect to $\boldsymbol{x}_0$

$$\begin{aligned}\frac{\partial \Phi(\boldsymbol{x}_0, R)}{\partial \boldsymbol{x}_0} \quad &= \quad -2 \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n \\ &\quad +2 \sum R^\top (\boldsymbol{y}_n - \boldsymbol{y}_0)\, p_n\end{aligned}$$

Set Derivative to Zero

- Set first derivative to zero: $\frac{\partial \Phi}{\partial \boldsymbol{x}_0} = 0$

$$0 \quad = \quad -2 \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n + 2 \sum R^\top (\boldsymbol{y}_n - \boldsymbol{y}_0)\, p_n$$

- This simplifies to

$$\sum (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n \quad = \quad R^\top \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)\, p_n$$

Set Derivative to Zero

- Set first derivative to zero: $\frac{\partial \Phi}{\partial \boldsymbol{x}_0} = 0$

$$0 \quad = \quad -2 \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n + 2 \sum R^\top (\boldsymbol{y}_n - \boldsymbol{y}_0)\, p_n$$

- This simplifies to

$$\sum (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n \quad = \quad R^\top \underline{\sum (\boldsymbol{y}_n - \boldsymbol{y}_0)\, p_n}$$

equal to zero as y_0 is the weighted mean of y_n

$$\Longrightarrow \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n \quad = \quad 0$$

Unknown $\boldsymbol{x}_0$ is the Weighted Mean of the Points to Transform

- As $\sum (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n \;=\; 0$
- We obtain $\sum \boldsymbol{x}_n\, p_n - \sum \boldsymbol{x}_0\, p_n \;=\; 0$
- This leads to

$$\boxed{\boldsymbol{x}_0 \;=\; \frac{\sum \boldsymbol{x}_n\, p_n}{\sum p_n}}$$

- The optimal value for $\boldsymbol{x}_0$ is the **weighted mean** of the points $\boldsymbol{x}_n$

Solve w.r.t. R

Compute R That Minimizes Φ

- Only the 3rd term of Φ depends on R

$$\begin{aligned}\Phi(\boldsymbol{x}_0, R) \quad = \quad & \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\top (\boldsymbol{y}_n - \boldsymbol{y}_0)\, p_n \\ & + \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)^\top (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n \\ & \boxed{-2 \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\top R (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n}\end{aligned}$$

- So we need to find R that maximizes

$$R^* = \underset{R}{\operatorname{argmax}} \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)^\top R (\boldsymbol{x}_n - \boldsymbol{x}_0)\, p_n$$

- with the constraint $R^\top R = I$

Exploit What We Know

- Given we know $\boldsymbol{x}_0$, compute mean-reduced coordinates as

$$\begin{aligned} \boldsymbol{a}_n &= (\boldsymbol{x}_n - \boldsymbol{x}_0) \\ \boldsymbol{b}_n &= (\boldsymbol{y}_n - \boldsymbol{y}_0) \end{aligned}$$

- This leads to the compact form

$$R^* = \underset{R}{\operatorname{argmax}} \sum \boldsymbol{b}_n^\top R \boldsymbol{a}_n \, p_n$$

Rewrite Using the Trace

- We can directly rewrite

$$R^* = \underset{R}{\operatorname{argmax}} \sum \boldsymbol{b}_n^\top R \boldsymbol{a}_n \, p_n$$

- using the trace as

$$R^* = \underset{R}{\operatorname{argmax}} \operatorname{tr}(RH)$$

- with the cross covariance matrix

$$H = \sum (\boldsymbol{a}_n \boldsymbol{b}_n^\top) \, p_n$$

- **Thus, find R that maximizes $\operatorname{tr}(RH)$**

Maximization Using SVD

- To find R that maximizes $\mathrm{tr}\,(RH)$, we can exploit the SVD
- SVD gives us

$$\mathrm{svd}(H) = UDV^\top$$

- with

$$U^\top U = I \qquad V^\top V = I \qquad D = \mathrm{diag}(d_i)$$

Maximization Using SVD

- Let's see what happens if we set

$$R = VU^\top$$

- Then, we obtain

$$\operatorname{tr}(RH) = \operatorname{tr}(\underbrace{VU^\top}_{R}\underbrace{UDV^\top}_{H}) = \operatorname{tr}(V\underbrace{U^\top U}_{I}DV^\top) = \operatorname{tr}(VDV^\top)$$

- and we can rewrite this as

$$\operatorname{tr}(VDV^\top) = \operatorname{tr}(VD^{\frac{1}{2}}D^{\frac{1}{2}}V^\top)$$

Maximization Using SVD

- As D is diagonal, we can write

$$\operatorname{tr}\left(VD^{\frac{1}{2}}D^{\frac{1}{2}}V^\top\right) = \operatorname{tr}\left(VD^{\frac{1}{2}}(D^{\frac{1}{2}}V)^\top\right)$$

- and with the definition $A = VD^{\frac{1}{2}}$

$$\operatorname{tr}(RH) = \operatorname{tr}\left(AA^\top\right)$$

- with A being a positive definite matrix (this results as V, D stem from SVD)

Exploit Inequality

- For every pos. definite matrix A holds

$$\operatorname{tr}\left(AA^\top\right) \geq \operatorname{tr}\left(R'AA^\top\right)$$

  for any rotation matrix R'
- Result of the Schwarz inequality
- This means

$$\operatorname{tr}(RH) = \operatorname{tr}\left(AA^\top\right) \geq \operatorname{tr}\left(R'AA^\top\right) = \operatorname{tr}(R'RH)$$

any other rotation matrix

- Thus, our choice $R = VU^\top$ was optimal as it maximizes the trace

Proof that $\mathrm{tr}\left(AA^\top\right) \geq \mathrm{tr}\left(R'AA^\top\right)$

optional

Lemma: For any positive definite matrix AA^t, and any orthonormal matrix B,

$$\text{Trace}\,(AA^t) \geq \text{Trace}\,(BAA^t).$$

Proof of Lemma: Let a_i be the ith column of A. Then

$$\begin{aligned}\text{Trace}\,(BAA^t) &= \text{Trace}\,(A^tBA)\\ &= \sum_i a_i^t(Ba_i).\end{aligned}$$

But, by the Schwarz inequality,

$$a_i^t(Ba_i) \leq \sqrt{(a_i^ta_i)(a_i^tB^tBa_i)} = a_i^ta_i.$$

Hence, $\text{Trace}\,(BAA^t) \leq \Sigma_i\, a_i^ta_i = \text{Trace}\,(AA^t)$. Q.E.D.

Let the SVD of H be:

See: Arun et al (1987) "Least-Squares Fitting of Two 3D Point Sets." IEEE T-PAMI 9(5), 698–700.

Optimal R

- The rotation matrix minimizing Φ is

$$\boxed{R = VU^\top}$$

- with $\operatorname{svd}(H) = UDV^\top$
- and $H = \sum (\boldsymbol{a}_n \boldsymbol{b}_n^\top)\, p_n$

Unique Solution?

- SVD provides the decomposition

  $$\mathrm{svd}(H) = UDV^\top$$

- The matrices U, V are 3 by 3 matrices
- U, V are rotation matrices
- Diagonal matrix $D = \mathrm{Diag}(\sigma_1, \sigma_2, \sigma_3)$
- Only if $\mathrm{rank}(H) = 3$, the rotation minimizing Φ is unique

Translation Vector

- Based on $\boldsymbol{x}_0$ and R, we can compute the translation vector $\boldsymbol{t}$ of our rigid body transformation
- Starting from

$$\boldsymbol{x}_0 = R^\top \boldsymbol{y}_0 - R^\top \boldsymbol{t}$$

- directly leads to

$$\boxed{\boldsymbol{t} = \boldsymbol{y}_o - R\boldsymbol{x}_0}$$

We Derived the Optimal Rigid Body Transform

- Rotation $R = VU^\top$
- Translation $\boldsymbol{t} = \boldsymbol{y}_0 - R\boldsymbol{x}_0$
- with

$$H = \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)(\boldsymbol{y}_n - \boldsymbol{y}_0)^\top p_n \qquad \mathrm{svd}(H) = UDV^\top$$

$$\boldsymbol{y}_0 = \frac{\sum \boldsymbol{y}_n p_n}{\sum p_n} \qquad \boldsymbol{x}_0 = \frac{\sum \boldsymbol{x}_n p_n}{\sum p_n}$$

We Derived the Optimal Rigid Body Transform

- The rigid body transformation with

$$\boxed{R = VU^\top \qquad \boldsymbol{t} = \boldsymbol{y}_0 - R\boldsymbol{x}_0}$$

- minimizes our objective function and thus $\sum ||\boldsymbol{y}_n - \bar{\boldsymbol{x}}_n||^2 p_n \rightarrow \min$

Note

- No initial guess needed (direct)
- No better solution exists (optimal)

Two Different Variants...

- There are two (sometimes confusing) variants of the problem formulation
- Variant 1:

$$H = \sum (\boldsymbol{x}_n - \boldsymbol{x}_0)(\boldsymbol{y}_n - \boldsymbol{y}_0)^\top p_n \qquad R = VU^\top$$

- Variant 2:

$$H = \sum (\boldsymbol{y}_n - \boldsymbol{y}_0)(\boldsymbol{x}_n - \boldsymbol{x}_0)^\top p_n \qquad R = UV^\top$$

- Both are equivalent!

Summary: Registration with Known Data Association

- Approach to compute the rigid body transformation between point clouds
- **Assumes known data association**
- Special case of the absolute orientation problem
- Efficient to implement
- Direct and optimal solution
- Effective and popular approach

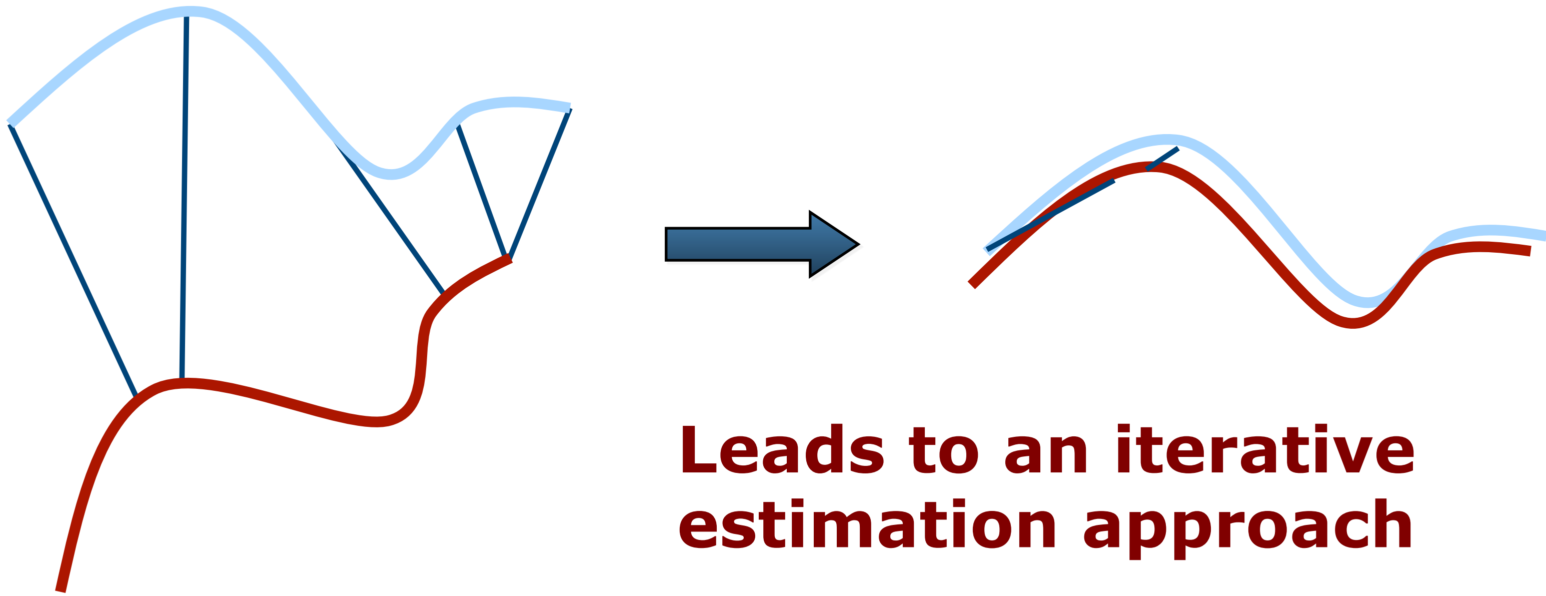
Outlook

Part 2
Point Cloud Registration with Unknown Data Association

ICP: Point Cloud Registration Estimating the Data Association

Outlook

If the correct correspondences are **not known**, it is generally impossible to determine the optimal rotation and translation in one step



Leads to an iterative estimation approach

Summary

- Registering point clouds is a central task in perception and mapping
- Compute the translation and rotation between point clouds or scans
- Given the correct data associations, the optimal transformation can be computed efficiently using SVD
- ICP (=standard registration algorithm) uses the solution discussed here

Further Reading

- Arun et al. “Least-Squares Fitting of Two 3D Point Sets”
- Besl & McKay “Registration of 3-D shapes”