Module 10: 3D Data Registration

Learning Objectives

- Learn the problem of 3D registration
- Learn Iterative Closest Point (ICP) algorithm
- Understand how numerical algorithms are developed to solve a real life problem

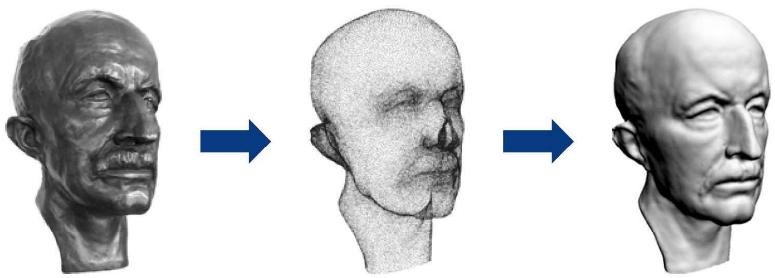
Sources

- Wiki: ICP
  https://en.wikipedia.org/wiki/Iterative_closest_point
- Additional literature:
  - Horn: *Closed-form solution of absolute orientation using unit quaternions*, Journal Opt.Soc.Amer. 4(4), 1987
  - P.Besl & N.McKay: *A method for registration of 3D shapes*, IEEE Trans. on PAMI, 1992
  - Y.Chen & G.Medioni: *Object modelling by registration of multiple range images*, Image Vision Comput. 10 (3): 145-155, 1991.
  - Z.Zhang, *Iterative point matching for registration of free-form curves and surfaces*, IJCV13 (12): 119–152, 1994.
  - Pulli: *Multiview registration for large data sets*, 3DIM 1999
  - Rusinkiewicz & Levoy: *Efficient variants of the ICP algorithm*, 3DIM 2001
  - Gelfand et al: *Geometrically stable sampling for the ICP algorithm*, 3DIM 2003

Outline

§1. Introduction
§2. ICP algorithm
§3. Mathematics behind ICP
§4. Summary

Surface Reconstruction



Physical model

Acquired point cloud

Reconstructed model

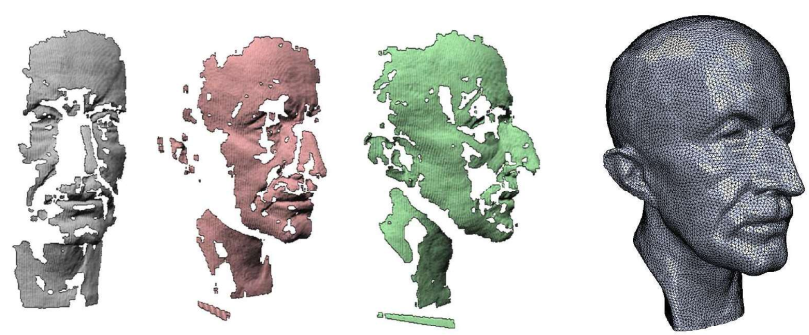
Example

Example (cont)

Upload to the web

Multiple Scans Needed

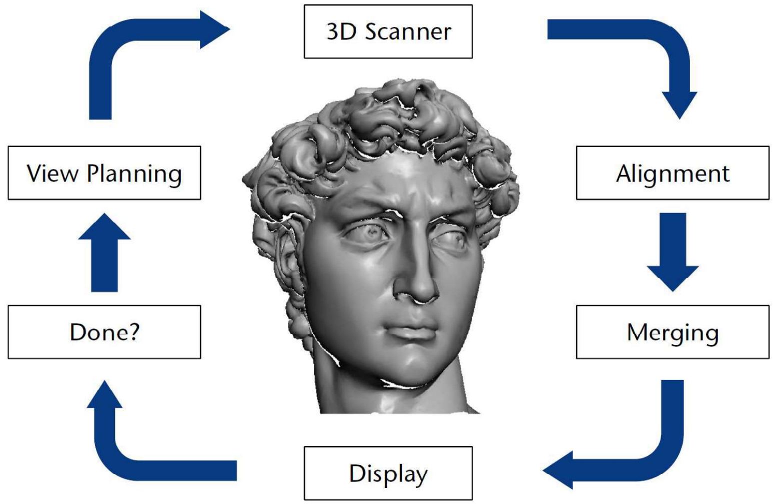
- Take snapshots from different directions
  - Requires scan alignment / registration



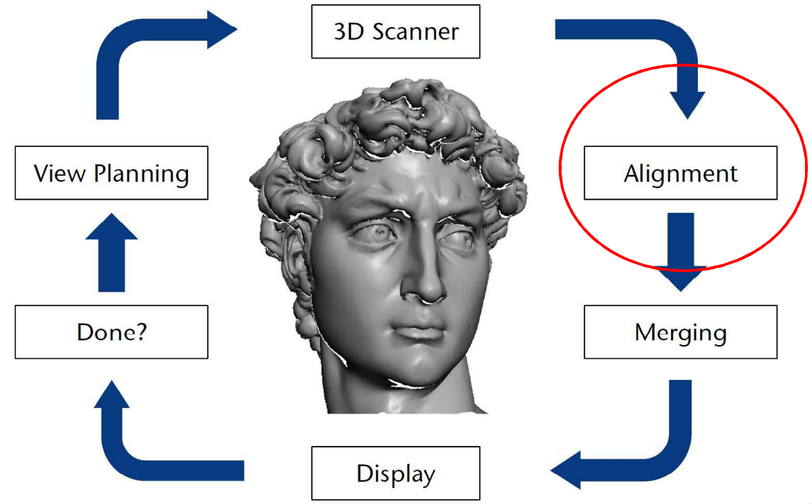
set of raw scans

reconstructed model

3D Model Acquisition Pipeline



A fundamental problem: 3D alignment



Outline

§1. Introduction

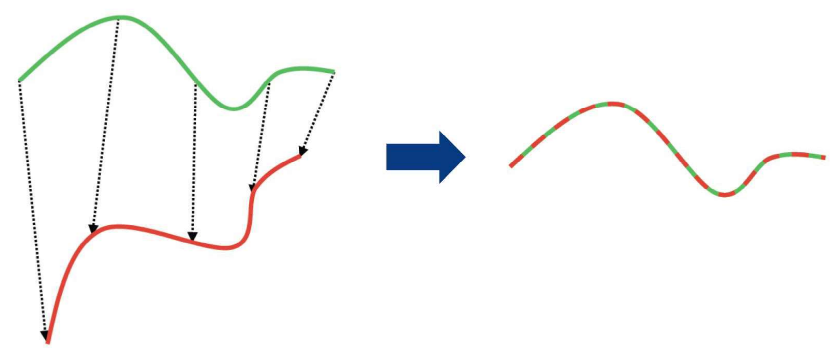
§2. ICP algorithm

§3. Mathematics behind ICP

§4. Summary

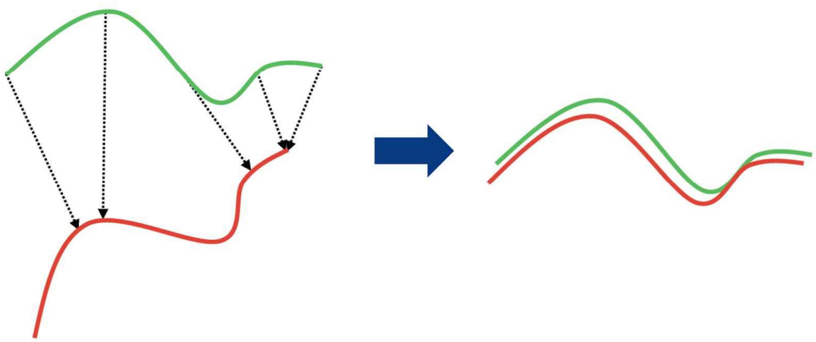
Aligning 3D Data

- If correct correspondences are known, one can find the correct relative rotation / translation
- How to find correspondences?



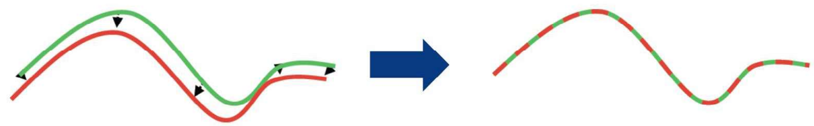
Aligning 3D Data (cont)

- Assume closest points correspond
- Iterate to find alignment



Aligning 3D Data (cont)

- Assume closest points correspond
- Iterate to find alignment
  - Iterative Closest Points (ICP) [Besl & McKay 92]
- Converges if starting position “close enough”



Basic ICP Algorithm

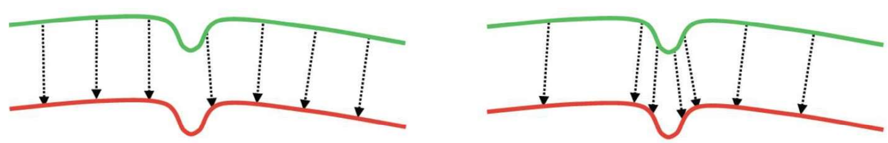
- Iterate until convergence
  1) select a subset of points p_i
  2) match each p_i to closest point q_i on other scan
  3) reject “bad” pairs (p_i, q_i)
  4) find rotation R and translation t to minimize

$$\min_{R,t} \sum_i \|p_i - Rq_i - t\|^2$$

  5) update scan alignment: $q_i \leftarrow Rq_i + t$

Step 1. Point Selection

- Use all points for matching
  - too complex
- Uniform / random subsampling
  - typically works well, but can miss features
- Stable sampling [Gelfand 2003]
  - normal-based sampling
  - slippage analysis



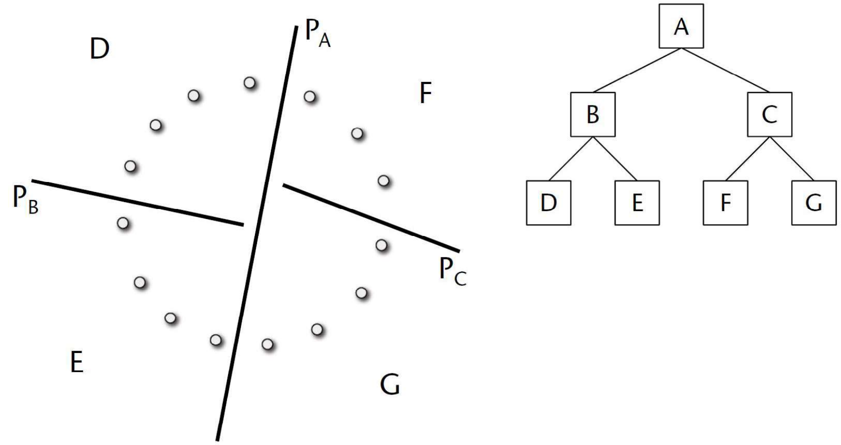
uniform sampling　　stable sampling

Step 2. Closest Point Search

- Find closest point of a query point
  - brute force: O(n) complexity
- Use hierarchical BSP tree
  - binary space partitioning tree (also kD-tree)
  - recursively partition 3D space by planes
  - tree should be balanced, put plane at median
  - log(n) tree levels, complexity O(log n)

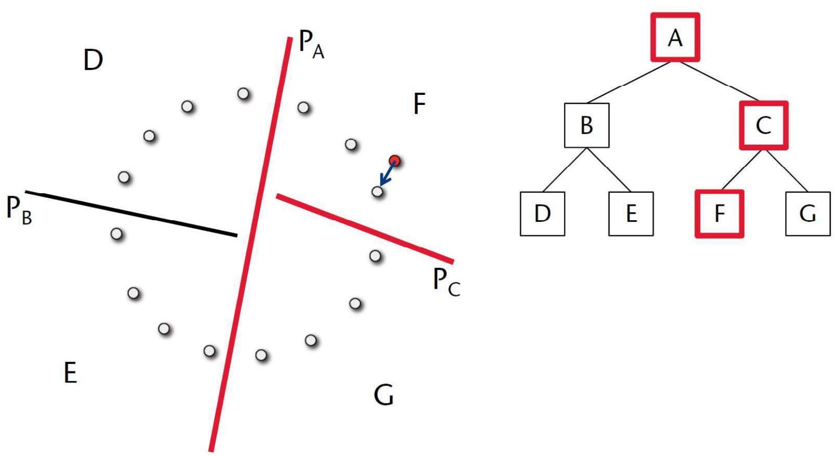
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

BSP Closest Points



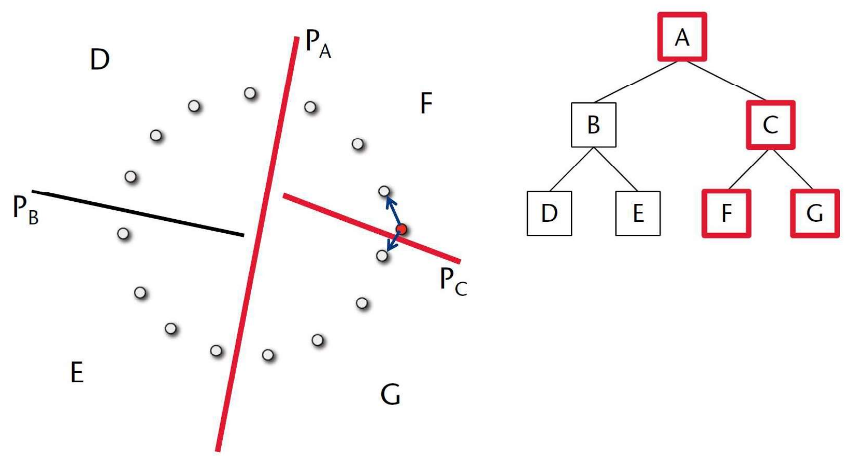
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

BSP Closest Points



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

BSP Closest Points



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

BSP Closest Points

```
BSPNode::dist(Point x, Scalar& dmin)
{
  if (leaf_node())
    for each sample point p[i]
      dmin = min(dmin, dist(x, p[i]));
  else
  {
    d = dist_to_plane(x);
    if (d < 0)
    {
      left_child->dist(x, dmin);
      if (|d| < dmin) right_child->dist(x, dmin);
    }
    else
    {
      right_child->dist(x, dmin);
      if (|d| < dmin) left_child->dist(x, dmin);
    }
  }
}
```


Step 3. Find Best Translation

- Define "barycentred" point sets

$$\bar{p} := \frac{1}{m}\sum_{i=1}^{m} p_i \qquad \bar{q} := \frac{1}{m}\sum_{i=1}^{m} q_i$$

$$\hat{p}_i := p_i - \bar{p} \qquad \hat{q}_i := q_i - \bar{q}$$

- Optimize translation vector t that maps barycentres onto each other

$$t = \bar{p} - R\bar{q}$$

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Step 4. Find Best Rotation

- Minimization problem simplifies to

$$\min_{R,t}\sum_i \|p_i - Rq_i - t\|^2 \quad \longrightarrow \quad \min_{R}\sum_i \|\hat{p}_i - R\hat{q}_i\|^2$$

- Consider the ***covariance matrix***

$$A = \sum_{i=1}^{m} \hat{p}_i \hat{q}_i^T \in \mathbb{R}^{3\times 3}$$

- ***Singular value decomposition*** of A gives best rotation:

$$A = U\Sigma V^T \quad \longrightarrow \quad R = UV^T$$

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Basic ICP Algorithm

- Iterate until convergence
  1) select a subset of points p_i
  2) match each p_i to closest point q_i on other scan
  3) reject "bad" pairs (p_i, q_i)
  4) find rotation R and translation t to minimize

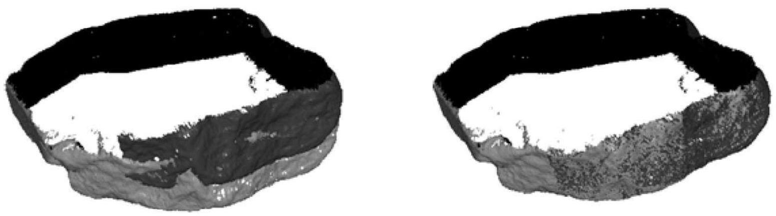
$$\min_{R,t}\sum_i \|p_i - Rq_i - t\|^2$$

  5) update scan alignment: $q_i \leftarrow Rq_i + t$

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Question for you?

- What if we want to align *n* >2 scans to each other?
- align each new scan to previous one?
  - no, leads to error accumulation
- instead, global registration



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Outline

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Basic problem

Suppose we are given two points sets

$$P = \{p_1, p_2, \cdots, p_m\}$$

and

$$Q = \{q_1, q_2, \cdots, q_m\}$$

in $\mathbf{R}^3$, and we are looking for the best *rigid tranform* to match them.

That is, we want to solve the optimization problem

$$\min_{R,t} \sum_{i=1}^{m} \|p_i - Rq_i - t\|^2$$

for some *rotation* $R \in \mathbf{R}^{3\times 3}$ and some *translation* $t \in \mathbf{R}^3$.

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Optimal Translation

Our first step is to eliminate the translation t from the problem. This can be achieved by shifting all points to their respective barycenters:

$$\bar{p} = \frac{1}{m}\sum_{i=1}^{m} p_i, \qquad \bar{q} = \frac{1}{m}\sum_{i=1}^{m} q_i$$

So the shifted points are

$$\hat{p}_i = p_i - \bar{p}, \qquad \hat{q}_i = q_i - \bar{q}.$$

It is clear that

$$\sum_{i=1}^{m} \hat{p}_i = 0 = \sum_{i=1}^{m} \hat{q}_i$$

and that

$$p_i - Rq_i - t = (\hat{P}_i + \bar{p}) - R(\hat{q}_i + \bar{q}) - t = \hat{p}_i - R\hat{q}_i - s$$

where $s = t - \bar{p} - R\bar{q}$.

Optimal Translation (cont)

Then,

$$\begin{aligned}
\sum_{i=1}^{m} \|p_i - Rq_i - t\|^2 &= \sum_{i=1}^{m} \|\hat{p}_i - R\hat{q}_i - s\|^2 \\
&= \sum_{i=1}^{m} \left((\hat{p}_i - R\hat{q}_i) - s\right)^T \left((\hat{p}_i - R\hat{q}_i) - s\right) \\
&= \sum_{i=1}^{m} (\hat{p}_i - R\hat{q}_i)^T (\hat{p}_i - R\hat{q}_i) - \sum_{i=1}^{m} s^T (\hat{p}_i - R\hat{q}_i) - \sum_{i=1}^{m} (\hat{p}_i - R\hat{q}_i)^T s + \sum_{i=1}^{m} s^T s \\
&= \sum_{i=1}^{m} \|\hat{p}_i - R\hat{q}_i\|^2 - 2s^T \left(\underbrace{\sum_{i=1}^{m} \hat{p}_i}_{=0} - R \underbrace{\sum_{i=1}^{m} \hat{q}_i}_{=0} \right) + m\|s\|^2 \\
&= \sum_{i=1}^{m} \|\hat{p}_i - R\hat{q}_i\|^2 + m\|s\|^2,
\end{aligned}$$

which is minimized if and only if $s = 0$, which gives the optimal translation:

$$t = \bar{p} - R\bar{q}.$$

Optimal Rotation

So we are left with the slightly simpler problem

$$\min_R \sum_{i=1}^{m} \|\hat{p}_i - R\hat{q}_i\|^2$$

for some rotation R. Noticing that $R^T R = I$ for any rotation matrix R, we get

$$\begin{aligned}
\sum_{i=1}^{m} \|\hat{p}_i - R\hat{q}_i\|^2 &= \sum_{i=1}^{m} (\hat{p}_i - R\hat{q}_i)^T (\hat{p}_i - R\hat{q}_i) \\
&= \sum_{i=1}^{m} \hat{p}_i{}^T \hat{p}_i - 2 \sum_{i=1}^{m} \hat{p}_i{}^T R\hat{q}_i + \sum_{i=1}^{m} \hat{q}_i{}^T R^T R\hat{q}_i \\
&= \sum_{i=1}^{m} \|\hat{p}_i\|^2 - 2 \sum_{i=1}^{m} \hat{p}_i{}^T R\hat{q}_i + \sum_{i=1}^{m} \|\hat{q}_i\|^2,
\end{aligned}$$

Hence the problem becomes solving

$$\max_R \sum_{i=1}^{m} \hat{p}_i^T R\hat{q}_i.$$

Basic algebra of matrices

We now use the simple tricks,

$$x^T y = \mathrm{trace}(xy^T)$$

for any vectors x and y, and

$$\mathrm{trace}(A+B) = \mathrm{trace}(A) + \mathrm{trace}(B), \qquad \mathrm{trace}(AB) = \mathrm{trace}(BA)$$

for any square matrices A and B, to see that

$$\sum_{i=1}^{m} \hat{p}_i{}^T R\hat{q}_i = \sum_{i=1}^{m} \mathrm{trace}(\hat{p}_i \hat{q}_i{}^T R^T) = \sum_{i=1}^{m} \mathrm{trace}(R^T \hat{p}_i \hat{q}_i{}^T) = \mathrm{trace}\left(R^T \sum_{i=1}^{m} \hat{p}_i \hat{q}_i{}^T \right).$$

Thus, by introducing the *covariance matrix*

$$A = \sum_{i=1}^{m} \hat{p}_i \hat{q}_i{}^T,$$

we arrive at the problem

$$\max_R \mathrm{trace}(R^T A).$$

Matrix decompositions

- Spectral decomposition

A symmetric real-valued matrix $M \in \mathbf{R}^{n \times n}$ can be written as

$$M = W \Lambda W^T$$

where

- $W \in \mathbf{R}^{n \times n}$ is the orthogonal matrix whose columns are the unit-length *eigenvectors* $\omega_1, \omega_2, \cdots, \omega_n$ of M;
- $\Lambda \in \mathbf{R}^{n \times n}$ is the diagonal matrix with the corresponding (real) *eigenvalues* $\lambda_1, \lambda_2, \cdots, \lambda_n$.

A rearrangement gives

$$M = (w_1, \ldots, w_n) \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix} \begin{pmatrix} w_1{}^T \\ \vdots \\ w_n{}^T \end{pmatrix} = (w_1, \ldots, w_n) \begin{pmatrix} \lambda_1 w_1{}^T \\ \vdots \\ \lambda_n w_n{}^T \end{pmatrix} = \sum_{k=1}^{n} w_k (\lambda_k w_k{}^T) = \sum_{k=1}^{n} \lambda_k w_k w_k{}^T.$$

Matrix decompositions

- Singular value decomposition (SVD)

A real-valued matrix $A \in \mathbf{R}^{n\times n}$ can be written as

$$A = U\Sigma V^T$$

where

- $U \in \mathbf{R}^{n\times n}$ and $V \in \mathbf{R}^{n\times n}$ are orthogonal matrices;
- $\Sigma \in \mathbf{R}^{n\times n}$ is the diagonal matrix that contains *singular values* $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$ of A.

Note that $A^T A$ is a symmetric matrix and

$$A^T A = (U\Sigma V^T)^T (U\Sigma V^T) = V\Sigma U^T U\Sigma V^T = V\Sigma^2 V^T.$$

We can see

- the columns of V are the unit-length eigenvectors of $A^T A$;
- the singular values σ_i of A are the square roots of the eigenvalues of $A^T A$.

Matrix decompositions

- Polar decomposition

A real-valued matrix $A \in \mathbf{R}^{n\times n}$ can have *polar decomposition*:

$$A = U\Sigma V^T = (UV^T)(V\Sigma V^T) = QS$$

where

- $Q = UV^T \in \mathbf{R}^{n\times n}$ is an orthogonal matrix;
- $S = V\Sigma V^T \in \mathbf{R}^{n\times n}$ is symmetric and positive semidefinite.

Moreover,

$$S^2 = S^T S = (V\Sigma V^T)(V\Sigma V^T) = V\Sigma^2 V^T = A^T A$$

$$S = V\Sigma V^T = \sum_{k=1}^{n} \sigma_k \omega_k \omega_k^T = \sum_{k=1}^{n} \sqrt{\lambda_k}\omega_k \omega_k^T$$

where $\omega_1, \cdots, \omega_n$ are the unit-length eigenvectors of $A^T A$, corresponding to their eigenvalues $\lambda_1, \cdots, \lambda_n$.

Optimal Rotation (cont)

Recall that our problem is $\boxed{\max_R \operatorname{trace}(R^T A)}$

We have

$$\operatorname{trace}(R^T A) = \operatorname{trace}(R^T QS) = \sum_{k=1}^{3} \sigma_k \operatorname{trace}(R^T Q\omega_k\omega_k^T)$$

Note that $\|\omega_k\| = 1$ and Q, R are both orthogonal. The *Cauchy-Schwarz inequality* gives

$$\operatorname{trace}(R^T Q w_k {w_k}^T) = \operatorname{trace}\left((R^T Q w_k){w_k}^T\right) = \left(R^T Q w_k\right)^T w_k$$

$$= {w_k}^T Q^T R w_k = \langle Q w_k, R w_k\rangle \leq \|Q w_k\|\|R w_k\| = 1$$

Therefore,

$$\operatorname{trace}(R^T A) = \operatorname{trace}(R^T QS) \leq \sum_{k=1}^{3} \sigma_k = \operatorname{trace}(\Sigma) = \operatorname{trace}(V\Sigma V^T) = \operatorname{trace}(S)$$

with equality if and only if $R^T Q = I$, and so the optimal rotation is

$$R = Q = UV^T.$$

Outline

Summary

- ICP algorithm
  - Optimal translation
  - Optimal rotation
- Least squares
- Eigenanalysis & matrix decomposition

End