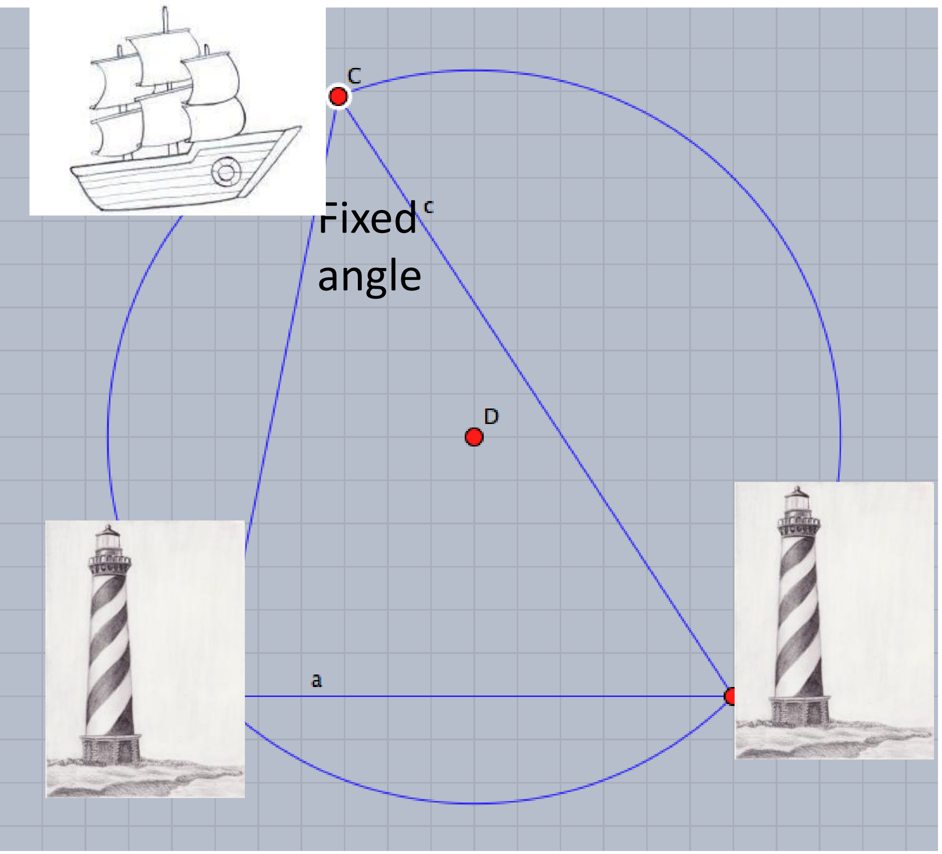
Robot Perception: Pose from Point Correspondences or the Perspective N Point Problem (PnP)

Kostas Daniilidis

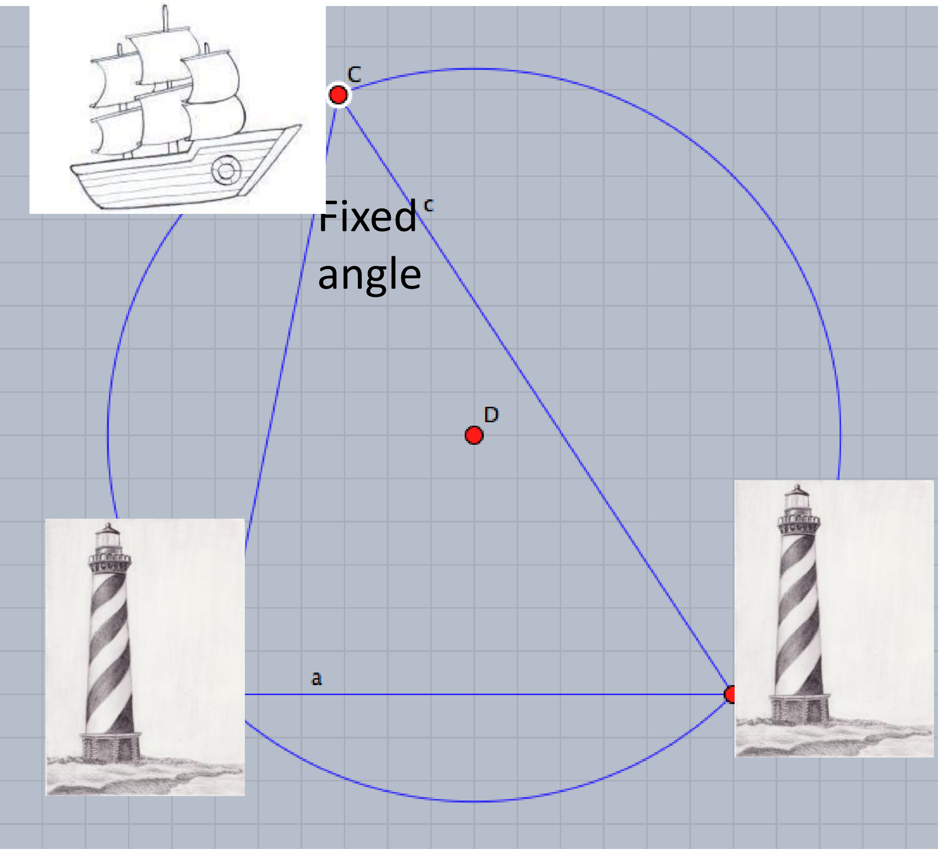
Where is the car if we know the projections of 3 points with known 3D coordinates?

Get back to an old idea of bearing!

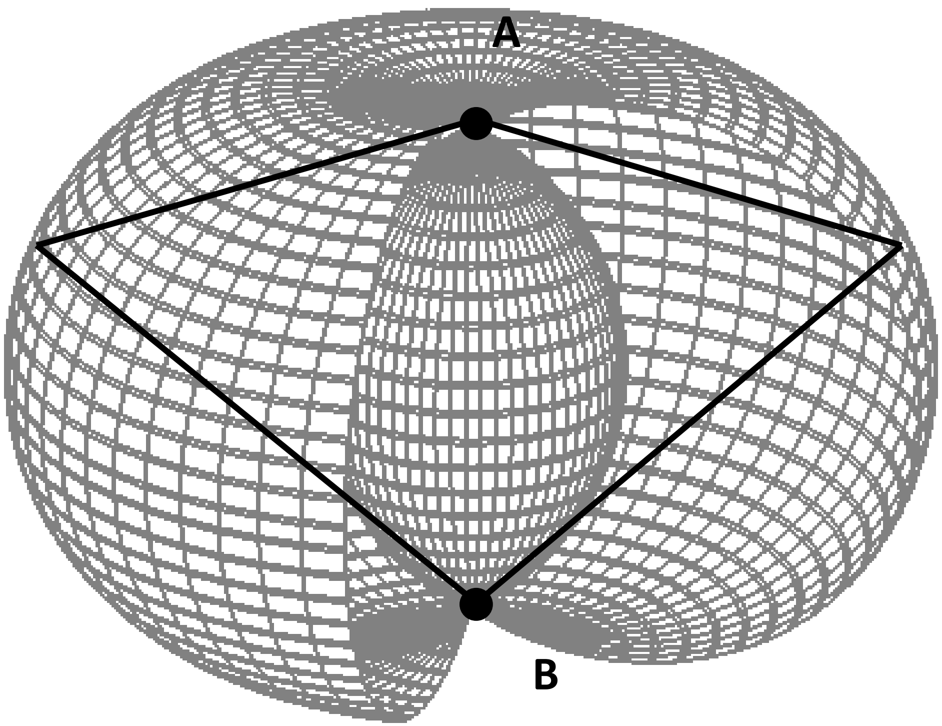
If we see two points like two lighthouses under a fixed angle, where am I?



Two points are not sufficient, camera can lie anywhere on a circle in 2D



… and on a toroid in 3D:



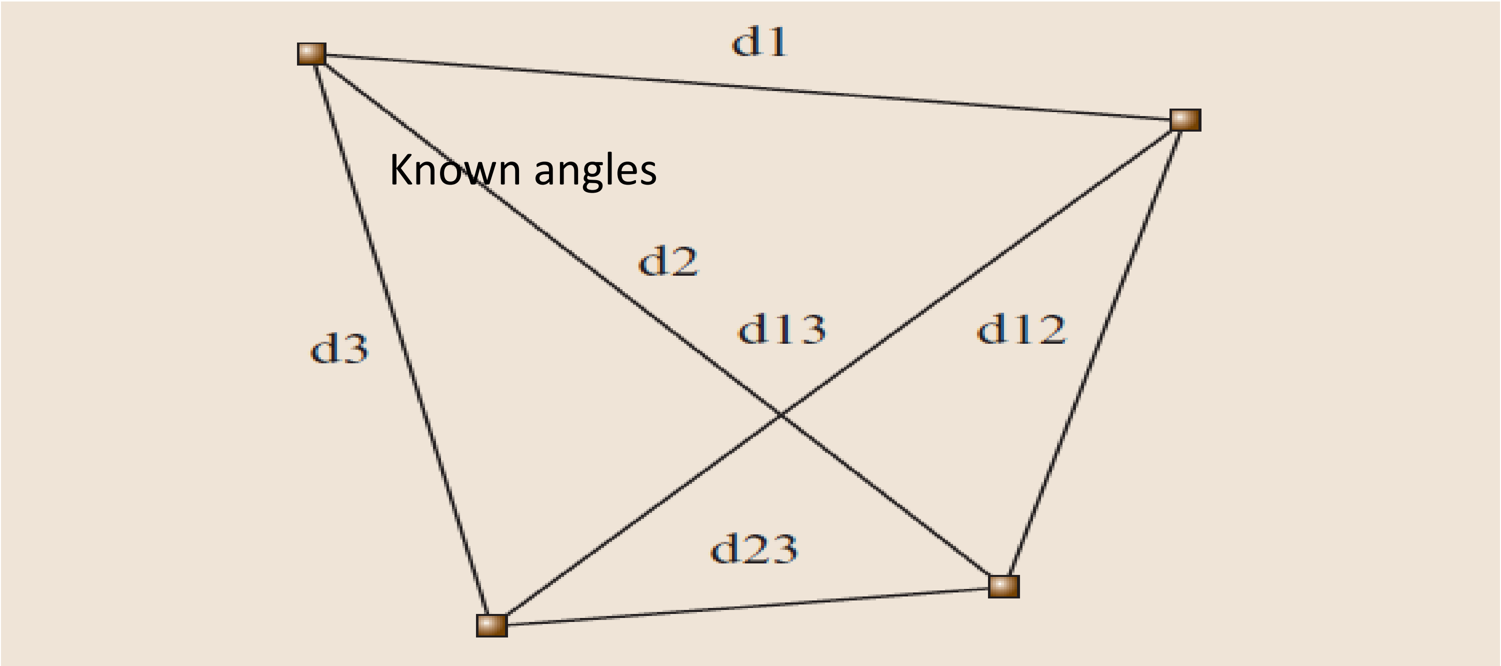
We need a 3rd point: The Perspective 3-Point problem of P3P

Or in Photogrammetry the Resection Problem:

The Snellius-Pothenot problem!

J. A. Grunert, "Das Pothenotische Problem in erweiterter Gestalt nebst Über seine Anwendungen in der Geodäsie, *Grunerts Archiv für Mathematik und Physik*, Band 1, 1841, pp. 238–248.

We need a 3rd point: The Perspective 3-Point problem of P3P



Let us assume two scene points and denote the known angle between their projections p_i and p_j as δ_{ij}

Let us denote the squared distance $\|P_i - P_j\|^2$ with d_{ij}^2 and the lengths of P_j with d_j.

Then cosine law reads

$$d_i^2 + d_j^2 - 2 d_i d_j \cos \delta_{ij} = d_{ij}^2$$

Pose from 3 points: The perspective 3-point problem (P3P)

$$\boxed{d_i^2 + d_j^2 - 2d_i d_j \cos\delta_{ij} = d_{ij}^2}$$

If we can recover d_i and d_j the rest will be an absolute orientation problem

$$\lambda_j p_j = \mathbf{R}P_j + T$$

to recover translation and rotation between camera and world coordinate system.

Pose from 3 points: The perspective 3-point problem (P3P)

$$d_i^2 + d_j^2 - 2d_i d_j \cos\delta_{ij} = d_{ij}^2$$

If we can recover d_i and d_j the rest will be an absolute orientation problem

$$\lambda_j p_j = \mathbf{R} P_j + T$$

to recover translation and rotation between camera and world coordinate system.

The cosine law

$$d_i^2 + d_j^2 - 2d_i d_j \cos\delta_{ij} = d_{ij}^2$$

applies for each point pair. With 3 points we could solve 3 quadratic equations for $d_{i=1\ldots3}$.

Set $d_2 = ud_1$ and $d_3 = vd_1$ and solve all three equations for d_1:

$$
\begin{aligned}
d_1^2 &= \frac{d_{23}^2}{u^2 + v^2 - 2uv\cos\delta_{23}} \\
d_1^2 &= \frac{d_{13}^2}{1 + v^2 - 2v\cos\delta_{13}} \\
d_1^2 &= \frac{d_{12}^2}{u^2 + 1 - 2u\cos\delta_{12}}
\end{aligned}
$$

which is equivalent to two quadratic equations in u and v.

The cosine law

$$d_i^2 + d_j^2 - 2d_i d_j \cos\delta_{ij} = d_{ij}^2$$

applies for each point pair. With 3 points we could solve 3 quadratic equations for $d_{i=1\ldots3}$.

Set $d_2 = ud_1$ and $d_3 = vd_1$ and solve all three equations for d_1:

$$\begin{aligned} d_1^2 &= \frac{d_{23}^2}{u^2 + v^2 - 2uv\cos\delta_{23}} \\ d_1^2 &= \frac{d_{13}^2}{1 + v^2 - 2v\cos\delta_{13}} \\ d_1^2 &= \frac{d_{12}^2}{u^2 + 1 - 2u\cos\delta_{12}} \end{aligned}$$

which is equivalent to two quadratic equations in u and v.

$$\begin{aligned} d_{13}^2(u^2 + v^2 - 2uv\cos\delta_{23}) &= d_{23}^2(1 + v^2 - 2v\cos\delta_{13}) \qquad (1)\\ d_{12}^2(1 + v^2 - 2v\cos\delta_{13}) &= d_{13}^2(u^2 + 1 - 2u\cos\delta_{12}) \qquad (2) \end{aligned}$$

We solve (1) for u^2,
we insert it in (2)
and solve for u which appears only linearly.
Then we insert u back in (1) and obtain a quartic (4th degree) in v which can have as many as 4 real solutions for v.

In the next lecture we will see how we can solve

$$P_i^c = RP_i + T$$

for R, T.

DIRECT SOLUTIONS OF THE PnP PROBLEM

Pose from N points in space given intrinsic parameters K and correspondences $(X_i, Y_i, Z_i, x_i, y_i)_{i=1\ldots N}$
where

$$\begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} \sim K^{-1} \begin{pmatrix} u_i \\ v_i \\ 1 \end{pmatrix}$$

where u_i, v_i are pixel coordinates.

Pose from N points in space given intrinsic parameters K and correspondences $(X_i, Y_i, Z_i, x_i, y_i)_{i=1\ldots N}$
where

$$\begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} \sim K^{-1} \begin{pmatrix} u_i \\ v_i \\ 1 \end{pmatrix}$$

where u_i, v_i are pixel coordinates.

Given $(X_i, Y_i, Z_i, x_i, y_i)_{i=1\ldots N}$
find R, T such that

$$\lambda_i \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} = R \begin{pmatrix} X_i \\ Y_i \\ Z_i \end{pmatrix} + T$$

How many points do we need?

$$
\begin{aligned}
x_i &= \frac{r_{11}X_i + r_{12}Y_i + r_{13}Z_i + T_1}{r_{31}X_i + r_{32}Y_i + r_{33}Z_i + T_3} \\
y_i &= \frac{r_{21}X_i + r_{22}Y_i + r_{23}Z_i + T_2}{r_{31}X_i + r_{32}Y_i + r_{33}Z_i + T_3}
\end{aligned}
$$

Assuming that point correspondences produce independent equations we would obtain $2N$ equations for N points.

The unknowns of a rigid transformation are 6, so it seems that the minimal number of points is 3.

A brute-force approach would be to eliminate depths λ_i and try to find R, T such that

$$\begin{aligned} x_i &= \frac{r_{11}X_i + r_{12}Y_i + r_{13}Z_i + T_1}{r_{31}X_i + r_{32}Y_i + r_{33}Z_i + T_3} \\ y_i &= \frac{r_{21}X_i + r_{22}Y_i + r_{23}Z_i + T_2}{r_{31}X_i + r_{32}Y_i + r_{33}Z_i + T_3} \end{aligned}$$

This is a non-linear problem which needs an iterative approach.

A "dirty" trick would be to solve for a vector of 12 unknowns (R', T') and then find the closest orthogonal R to R'.