ECE276A: Sensing & Estimation in Robotics
Lecture 5: Localization and Odometry from Point Features

Nikolay Atanasov

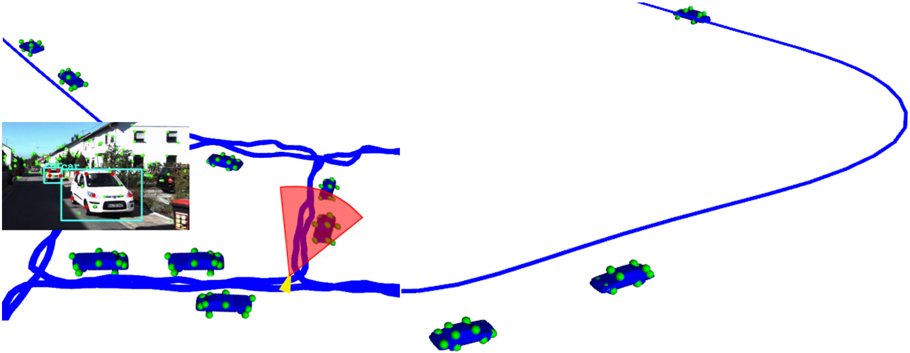
natanasov@ucsd.edu

UC San Diego
JACOBS SCHOOL OF ENGINEERING
Electrical and Computer Engineering

Outline

Simultaneous Localization and Mapping (SLAM)

- SLAM is a fundamental problem for mobile robot autonomy
- Basic information necessary to perform any robot task:
  - Where am I? $\Rightarrow$ **Localization**
  - What is around me? $\Rightarrow$ **Mapping**
- SLAM problem: given sensor measurements $\mathbf{z}_{0:T}$ (e.g., images) and control inputs $\mathbf{u}_{0:T-1}$ (e.g., velocity), estimate the robot state trajectory $\mathbf{x}_{0:T}$ (e.g., pose) and build a map $\mathbf{m}$ of the environment



Mathematical Formulation of SLAM Problems

- **Mapping**: given robot state trajectory $\mathbf{x}_{0:T}$ and sensor measurements $\mathbf{z}_{0:T}$ with observation model h, build a map $\mathbf{m}$ of the environment

$$\min_{\mathbf{m}} \sum_{t=0}^{T} \|\mathbf{z}_t - h(\mathbf{x}_t, \mathbf{m})\|_2^2$$

- **Localization**: given a map $\mathbf{m}$ of the environment, sensor measurements $\mathbf{z}_{0:T}$ with observation model h, and control inputs $\mathbf{u}_{0:T-1}$ with motion model f, estimate the robot state trajectory $\mathbf{x}_{0:T}$

$$\min_{\mathbf{x}_{0:T}} \sum_{t=0}^{T} \|\mathbf{z}_t - h(\mathbf{x}_t, \mathbf{m})\|_2^2 + \sum_{t=0}^{T-1} \|\mathbf{x}_{t+1} - f(\mathbf{x}_t, \mathbf{u}_t)\|_2^2$$

- **SLAM**: given initial robot state $\mathbf{x}_0$, sensor measurements $\mathbf{z}_{1:T}$ with observation model h, and control inputs $\mathbf{u}_{0:T-1}$ with motion model f, estimate the robot state trajectory $\mathbf{x}_{1:T}$ and build a map $\mathbf{m}$

$$\min_{\mathbf{x}_{1:T}, \mathbf{m}} \sum_{t=1}^{T} \|\mathbf{z}_t - h(\mathbf{x}_t, \mathbf{m})\|_2^2 + \sum_{t=0}^{T-1} \|\mathbf{x}_{t+1} - f(\mathbf{x}_t, \mathbf{u}_t)\|_2^2$$

Example: Localization with Linear Models

- State: $\mathbf{x}_t \in \mathbb{R}^n$
- Motion model: $\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t) = F\mathbf{x}_t + G\mathbf{u}_t$
- Observation model: $\mathbf{z}_t = h(\mathbf{x}_t) = H\mathbf{x}_t$
- Localization: given $\mathbf{x}_0 = \mathbf{0}$, sensor measurements $\mathbf{z}_{1:T}$, and control inputs $\mathbf{u}_{0:T-1}$, estimate the state trajectory $\mathbf{x}_{1:T}$

$$\min_{\mathbf{x}_{1:T}} c(\mathbf{x}_{1:T}) := \sum_{t=1}^{T} \|\mathbf{z}_t - H\mathbf{x}_t\|_2^2 + \sum_{t=0}^{T-1} \|\mathbf{x}_{t+1} - F\mathbf{x}_t - G\mathbf{u}_t\|_2^2$$

- Gradient descent: initialize $\mathbf{x}_{1:T}^{(0)}$ and iterate:

$$\mathbf{x}_{1:T}^{(k+1)} = \mathbf{x}_{1:T}^{(k)} - \alpha^{(k)} \nabla c(\mathbf{x}_{1:T}^{(k)})$$

Example: Localization with Linear Models

- $\left\| \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \right\|_2^2 = \|x_1 - y_1\|_2^2 + \|x_2 - y_2\|_2^2$ for $x_1, y_1 \in \mathbb{R}^{d_1}$, $x_2, y_2 \in \mathbb{R}^{d_2}$
- Express the least-squares localization problem in matrix notation:

$$
\begin{aligned}
c(\mathbf{x}_{1:T}) &= \sum_{t=1}^{T} \|\mathbf{z}_t - H\mathbf{x}_t\|_2^2 + \sum_{t=0}^{T-1} \|\mathbf{x}_{t+1} - F\mathbf{x}_t - G\mathbf{u}_t\|_2^2 \\
&= \left\| \begin{bmatrix} \mathbf{z}_1 - H\mathbf{x}_1 \\ \vdots \\ \mathbf{z}_T - H\mathbf{x}_T \end{bmatrix} \right\|_2^2 + \left\| \begin{bmatrix} \mathbf{x}_1 - F\mathbf{x}_0 - G\mathbf{u}_0 \\ \vdots \\ \mathbf{x}_T - F\mathbf{x}_{T-1} - G\mathbf{u}_{T-1} \end{bmatrix} \right\|_2^2 \\
&= \left\| \begin{bmatrix} H & & \\ & \ddots & \\ & & H \end{bmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_T \end{pmatrix} - \begin{bmatrix} \mathbf{z}_1 \\ \vdots \\ \mathbf{z}_T \end{bmatrix} \right\|_2^2 + \left\| \begin{bmatrix} -I & & & \\ F & \ddots & & \\ & \ddots & \ddots & \\ & & F & -I \end{bmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_T \end{pmatrix} + \begin{bmatrix} F\mathbf{x}_0 + G\mathbf{u}_0 \\ G\mathbf{u}_1 \\ \vdots \\ G\mathbf{u}_{T-1} \end{bmatrix} \right\|_2^2
\end{aligned}
$$

Example: Localization with Linear Models

- Objective:

$$c(\mathbf{x}_{1:T}) = \left\| \begin{bmatrix} H & & & \\ & \ddots & & \\ & & \ddots & \\ & & & H \\ -I & & & \\ F & \ddots & & \\ & \ddots & \ddots & \\ & & F & -I \end{bmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_T \end{pmatrix} + \begin{bmatrix} -\mathbf{z}_1 \\ \vdots \\ -\mathbf{z}_T \\ F\mathbf{x}_0 + G\mathbf{u}_0 \\ G\mathbf{u}_1 \\ \vdots \\ G\mathbf{u}_{T-1} \end{bmatrix} \right\|_2^2$$

$$= \|A\mathbf{x}_{1:T} + \mathbf{b}\|_2^2$$

- Gradient:

$$\nabla c(\mathbf{x}_{1:T}) = 2A^\top(A\mathbf{x}_{1:T} + \mathbf{b})$$

- Gradient descent: initialize $\mathbf{x}_{1:T}^{(0)}$ and iterate:

$$\mathbf{x}_{1:T}^{(k+1)} = \mathbf{x}_{1:T}^{(k)} - 2\alpha^{(k)} A^\top(A\mathbf{x}_{1:T}^{(k)} + \mathbf{b})$$

Project 1: Orientation Tracking

- Consider a rigid body undergoing pure rotation
- **State**: orientation $\mathbf{q}_t \in \mathbb{H}_*$ of the body frame relative to the world frame
- **Control**: body-frame angular velocity $\mathbf{u}_t \in \mathbb{R}^3$ obtained from gyroscope measurements in rad/sec during time interval τ_t
- **Motion model**: $\mathbf{q}_{t+1} = f(\mathbf{q}_t, \tau_t \mathbf{u}_t) := \mathbf{q}_t \circ \exp([0,\ \tau_t \mathbf{u}_t/2])$
- **Observation model**: body-frame acceleration $\mathbf{z}_t \in \mathbf{R}^3$ obtained from accelerometer measurements in m/sec^2 should approximately match the world-frame gravity acceleration $-g\mathbf{e}_3$:

$$\mathbf{z}_t = h(\mathbf{q}_t) := \bar{\mathbf{q}}_t \circ [0,\ -g\mathbf{e}_3] \circ \mathbf{q}_t$$

Project 1: Orientation Tracking

- Starting with $\mathbf{q}_0 = [1, \mathbf{0}] \in \mathbb{H}_*$, formulate an optimization problem to estimate $\mathbf{q}_{1:T}$ using the gyroscope inputs $\mathbf{u}_{0:T-1}$ and accelerometer measurements $\mathbf{z}_{1:T}$
- The optimization problem is **constrained** because we require that $\mathbf{q}_t$ is a valid orientation, i.e., $\mathbf{q}_t \in \mathbb{H}_*$:

$$\begin{aligned} \min_{\mathbf{q}_{1:T}} \;\; & c(\mathbf{q}_{1:T}) := \sum_{t=1}^{T} \|\mathbf{z}_t - h(\mathbf{q}_t)\|_2^2 + \sum_{t=0}^{T-1} \|2 \log\left(\mathbf{q}_{t+1}^{-1} \circ f(\mathbf{q}, \tau_t \mathbf{u}_t)\right)\|_2^2 \\ \text{s.t.} \;\; & \|\mathbf{q}_t\|_2 = 1, \;\; \forall t \end{aligned}$$

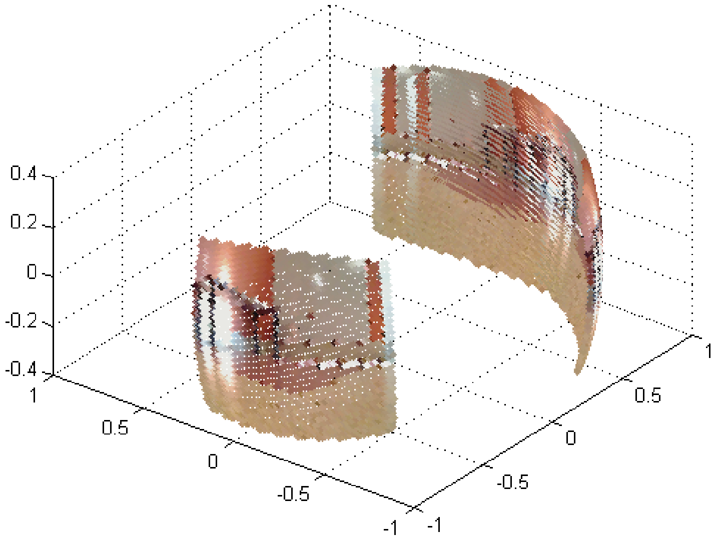
- **Possible approach**: projected gradient descent

$$\mathbf{q}_{1:T}^{(k+1)} = \Pi_{\mathbb{H}_*}\left(\mathbf{q}_{1:T}^{(k)} - \alpha^{(k)} \nabla c(\mathbf{q}_{1:T}^{(k)})\right)$$

Project 1: Panorama

- **Input**: image I and camera-to-world orientation R
- Suppose the image lies on a sphere and compute the world coordinates of each pixel:
  1. Find longitude (λ) and latitude (ϕ) of each pixel using the number of rows and columns and the horizontal (60°) and vertical (45°) fields of view
  2. Convert spherical $(\lambda, \phi, 1)$ to Cartesian coordinates assuming depth 1
  3. Rotate the Cartesian coordinates to the world frame using R
- Project world pixel coordinates to a cylinder and unwrap:
  1. Convert Cartesian to spherical coordinates
  2. Inscribe the sphere in a cylinder so that a point $(\lambda, \phi, 1)$ on the sphere has height ϕ on the cylinder and longitude λ along the cylinder circumference
  3. Unwrap the cylinder surface to a rectangular image with width 2π radians and height π radians
  4. Different options for sphere to plane projection: equidistant, equal area, Miller, etc. (see `https://en.wikipedia.org/wiki/List_of_map_projections`)

Project 1: Panorama



Outline

Localization and Odometry from Point Features

- **Point-cloud map**: suppose the map is represented as a set $\{\mathbf{m}_i\}_i$ of points $\mathbf{m}_i \in \mathbb{R}^d$
- **Observation model**: relates an observation $\mathbf{z}_i$ obtained from robot position $\mathbf{p}$ and orientation θ or R with the point $\mathbf{m}_i$ that generated it:
  - **Position Sensor**: $\mathbf{z}_i = R^\top(\mathbf{m}_i - \mathbf{p})$
  - **Range Sensor**: $\mathbf{z}_i = \|\mathbf{m}_i - \mathbf{p}\|_2$
  - **Bearing Sensor**: $\mathbf{z}_i = \arctan\left(\frac{m_{i,y} - p_y}{m_{i,x} - p_x}\right) - \theta$
  - **Camera Sensor**: $\mathbf{z}_i = K\pi\left(R^\top(\mathbf{m}_i - \mathbf{p})\right)$
- **Localization Problem**: Given landmark positions $\{\mathbf{m}_i\}_i$ and measurements $\{\mathbf{z}_i\}_i$ at one time instance, determine the global robot position $\mathbf{p}$ and orientation θ or R
- **Odometry Problem**: Given measurements $\mathbf{z}_{t,i}$, $\mathbf{z}_{t+1,i}$ at two time instances, determine the relative position ${}_t\mathbf{p}_{t+1}$ and orientation ${}_t\theta_{t+1}$ or ${}_tR_{t+1}$ between the two robot frames at time t and $t+1$

2-D Localization from Relative Position Measurements

- **Goal**: determine the robot position $\mathbf{p} \in \mathbb{R}^2$ and orientation $\theta \in (-\pi, \pi]$
- **Given**: two landmark positions $\mathbf{m}_1, \mathbf{m}_2 \in \mathbb{R}^2$ (world frame) and **relative position** measurements (body frame):

$$\mathbf{z}_i = R^\top(\theta)(\mathbf{m}_i - \mathbf{p}) \in \mathbb{R}^2, \quad i = 1, 2$$

- Let $\delta\mathbf{z} := \mathbf{z}_1 - \mathbf{z}_2$ and $J := \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ so that:

$$\mathbf{m}_1 - \mathbf{m}_2 = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} (\mathbf{z}_1 - \mathbf{z}_2) = \begin{bmatrix} \delta\mathbf{z} & J\delta\mathbf{z} \end{bmatrix} \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$$

- As long as $\det \begin{bmatrix} \delta\mathbf{z} & J\delta\mathbf{z} \end{bmatrix} = \|\delta\mathbf{z}\|_2^2 = \|\mathbf{m}_1 - \mathbf{m}_2\|_2^2 \neq 0$, we can compute:

$$\begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} = \frac{1}{\|\delta\mathbf{z}\|_2^2} \begin{bmatrix} \delta z_x & \delta z_y \\ -\delta z_y & \delta z_x \end{bmatrix} (\mathbf{m}_1 - \mathbf{m}_2) \quad \Rightarrow \quad \theta = \mathbf{atan2}(\sin\theta, \cos\theta)$$

- Given the orientation θ, we can then obtain the robot position:

$$\mathbf{p} = \frac{1}{2}\left((\mathbf{m}_1 + \mathbf{m}_2) - R(\theta)(\mathbf{z}_1 + \mathbf{z}_2)\right)$$

3-D Localization from Relative Position Measurements

- **Goal**: determine the robot position $\mathbf{p} \in \mathbb{R}^3$ and orientation $R \in SO(3)$
- **Given**: three landmark positions $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3 \in \mathbb{R}^3$ (world frame) and **relative position** measurements (body frame):

$$\mathbf{z}_i = R^\top(\mathbf{m}_i - \mathbf{p}) \in \mathbb{R}^3, \quad i = 1, 2, 3$$

- Let $\mathbf{m}_{ij} := \mathbf{m}_i - \mathbf{m}_j$ and $\mathbf{z}_{ij} = \mathbf{z}_i - \mathbf{z}_j$ and compute:

$$\mathbf{m}_{12} \times \mathbf{m}_{13} = (R\mathbf{z}_{12}) \times (R\mathbf{z}_{13}) = R(\mathbf{z}_{12} \times \mathbf{z}_{13})$$

- The vector $\mathbf{m}_{12} \times \mathbf{m}_{13}$ provides orthogonal information to $\mathbf{m}_1$ and $\mathbf{m}_2$ and can be used to estimate the orientation R **as long as the three points are not all on the same line**:

$$\begin{bmatrix}\mathbf{m}_{12} & \mathbf{m}_{13} & \mathbf{m}_{12} \times \mathbf{m}_{13}\end{bmatrix} = R\begin{bmatrix}\mathbf{z}_{12} & \mathbf{z}_{13} & \mathbf{z}_{12} \times \mathbf{z}_{13}\end{bmatrix}$$
$$R = \begin{bmatrix}\mathbf{m}_{12} & \mathbf{m}_{13} & \mathbf{m}_{12} \times \mathbf{m}_{13}\end{bmatrix}\begin{bmatrix}\mathbf{z}_{12} & \mathbf{z}_{13} & \mathbf{z}_{12} \times \mathbf{z}_{13}\end{bmatrix}^{-1}$$

- Given the orientation R, we can then obtain the robot position:

$$\mathbf{p} = \frac{1}{3}\sum_{i=1}^{3}(\mathbf{m}_i - R\mathbf{z}_i)$$

3-D Localization from Relative Position Measurements

- **Goal**: determine the robot position $\mathbf{p} \in \mathbb{R}^3$ and orientation $R \in SO(3)$
- **Given**: n landmark positions $\mathbf{m}_i \in \mathbb{R}^3$ (world frame) and **relative position** measurements (body frame):

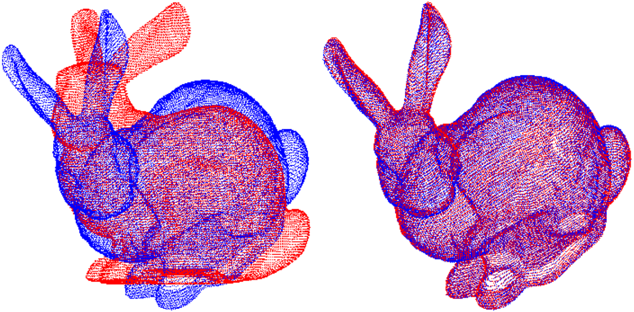
$$\mathbf{z}_i = R^\top (\mathbf{m}_i - \mathbf{p}) \in \mathbb{R}^3, \quad i = 1, \ldots, n$$

- Localization from relative position measurements is known as the **point cloud registration** problem
- Given two sets $\{\mathbf{m}_i\}$ and $\{\mathbf{z}_j\}$ of points, find the transformation $\mathbf{p}$, R that aligns them
- The **data association** $\Delta := \{(i, j) : \mathbf{m}_i \text{ corresponds to } \mathbf{z}_j\}$ that specifies which observation j corresponds to landmark i might not be available

Point Cloud Registration

- Given two sets $\{\mathbf{m}_i\}$ and $\{\mathbf{z}_j\}$ of points in $\mathbb{R}^d$, find the transformation $\mathbf{p} \in \mathbb{R}^d$, $R \in SO(d)$ and data association Δ that align them:

$$\min_{R \in SO(d), \mathbf{p} \in \mathbb{R}^d, \Delta} f(R, \mathbf{p}, \Delta) := \sum_{(i,j) \in \Delta} w_{ij} \|(R\mathbf{z}_j + \mathbf{p}) - \mathbf{m}_i\|_2^2$$



Known Data Association: Kabsch Algorithm

- Find the transformation $\mathbf{p} \in \mathbb{R}^d$, $R \in SO(d)$ between sets $\{\mathbf{m}_i\}$ and $\{\mathbf{z}_i\}$ of associated points:

$$\min_{R \in SO(d), \mathbf{p} \in \mathbb{R}^d} f(R, \mathbf{p}) := \sum_i w_i \|(R\mathbf{z}_i + \mathbf{p}) - \mathbf{m}_i\|_2^2$$

- The optimal translation is obtained by setting $\nabla_{\mathbf{p}} f(R, \mathbf{p})$ to zero:

$$\mathbf{0} = \nabla_{\mathbf{p}} f(R, \mathbf{p}) = 2 \sum_i w_i \left((R\mathbf{z}_i + \mathbf{p}) - \mathbf{m}_i\right)$$

- Let the point cloud centroids be:

$$\bar{\mathbf{m}} := \frac{\sum_i w_i \mathbf{m}_i}{\sum_i w_i} \qquad \bar{\mathbf{z}} := \frac{\sum_i w_i \mathbf{z}_i}{\sum_i w_i}$$

- Solving $\nabla_{\mathbf{p}} f(R, \mathbf{p}) = \mathbf{0}$ for $\mathbf{p}$ leads to:

$$\mathbf{p} = \bar{\mathbf{m}} - R\bar{\mathbf{z}}$$

Known Data Association: Kabsch Algorithm

- Replace $\mathbf{p} = \bar{\mathbf{m}} - R\bar{\mathbf{z}}$ in $f(R, \mathbf{p})$:

$$f(R, \bar{\mathbf{m}} - R\bar{\mathbf{z}}) = \sum_i w_i \| R(\mathbf{z}_i - \bar{\mathbf{z}}) - (\mathbf{m}_i - \bar{\mathbf{m}}) \|_2^2$$

- Define the centered point clouds:

$$\delta\mathbf{m}_i := \mathbf{m}_i - \bar{\mathbf{m}} \qquad\qquad \delta\mathbf{z}_i := \mathbf{z}_i - \bar{\mathbf{z}}$$

- Finding the optimal rotation reduces to:

$$\min_{R \in SO(d)} \sum_i w_i \| R\delta\mathbf{z}_i - \delta\mathbf{m}_i \|_2^2$$

- The objective function can be simplified further:

$$\sum_i w_i \| R\delta\mathbf{z}_i - \delta\mathbf{m}_i \|_2^2 = \sum_i w_i \left(\delta\mathbf{z}_i^\top \underbrace{R^\top R}_{I} \delta\mathbf{z}_i - 2\delta\mathbf{m}_i^\top R\delta\mathbf{z}_i + \delta\mathbf{m}_i^\top \delta\mathbf{m}_i \right)$$

- Note that:
  - $\delta\mathbf{z}_i^\top \delta\mathbf{z}_i$ and $\delta\mathbf{m}_i^\top \delta\mathbf{m}_i$ are constant wrt R
  - $\sum_i w_i \delta\mathbf{m}_i^\top R\delta\mathbf{z}_i = \sum_i w_i \operatorname{tr}(\delta\mathbf{m}_i^\top R\delta\mathbf{z}_i) = \operatorname{tr}\left(\left(\sum_i w_i \delta\mathbf{z}_i \delta\mathbf{m}_i^\top \right) R \right)$

Known Data Association: Kabsch Algorithm

- **Wahba's problem**: to determine the rotation R that aligns two associated centered point clouds $\{\delta\mathbf{m}_i\}$ and $\{\delta\mathbf{z}_i\}$, we need to solve a linear optimization problem in $SO(d)$:

$$\max_{R \in SO(d)} \operatorname{tr}(Q^\top R)$$

  where $Q := \sum_i w_i \delta\mathbf{m}_i \delta\mathbf{z}_i^\top$

- Wahba's problem can be solved via the **Kabsch algorithm**

Known Data Association: Kabsch Algorithm

- **Wahba's problem**: $\max_{R \in SO(d)} \operatorname{tr}\left(Q^\top R\right)$
- **SVD**: let $Q = U\Sigma V^\top$ be the singular value decomposition of Q
- The singular vectors U, V and singular values Σ satisfy:

$$\Sigma_{ii} \geq 0 \qquad U^\top U = I \qquad \det(U) = \pm 1 \qquad V^\top V = I \qquad \det(V) = \pm 1$$

- Let $W := U^\top R V$ such that $W^\top W = I$ and $\det(W) = \pm 1$
- The columns $\mathbf{w}_j$ of W are orthonormal, $\mathbf{w}_j^\top \mathbf{w}_j = 1$, and hence:

$$1 = \mathbf{w}_j^\top \mathbf{w}_j = \sum_i W_{ij}^2 \qquad \Rightarrow \qquad W_{ij}^2 \leq 1 \qquad \Rightarrow \qquad |W_{ij}| \leq 1$$

- Since Σ is diagonal with $\Sigma_{ii} \geq 0$:

$$\operatorname{tr}(Q^\top R) = \operatorname{tr}(\Sigma U^\top R V) = \operatorname{tr}(\Sigma W) = \sum_i \Sigma_{ii} W_{ii} \leq \sum_i \Sigma_{ii}$$

- The maximum is achieved with $W = I$:

$$W = I \quad \Rightarrow \quad U^\top R V = I \quad \underset{\text{reflection}}{\overset{\text{avoids}}{\Rightarrow}} \quad R = U \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & 1 & \\ & & & \det(UV^\top) \end{bmatrix} V^\top$$

Unknown Data Association: Iterative Closest Point (ICP)

- Find the transformation $\mathbf{p}$, R between sets $\{\mathbf{m}_i\}$ and $\{\mathbf{z}_j\}$ of points with **unknown** data association Δ

- **ICP algorithm**: iterates between finding associations Δ based on **closest points** and applying the **Kabsch algorithm** to determine $\mathbf{p}$, R

- Initialize with $\mathbf{p}_0$, R_0 (**sensitive to initial guess**) and iterate
  1. Given $\mathbf{p}_k$, R_k, find correspondences $(i, j) \in \Delta$ based on **closest points**:
  $$i \quad \leftrightarrow \quad \arg\min_j \|\mathbf{m}_i - (R_k \mathbf{z}_j + \mathbf{p}_k)\|_2^2$$
  2. Given correspondences $(i, j) \in \Delta$, find $\mathbf{p}_{k+1}$, R_{k+1} via **Kabsch algorithm**

Unknown Data Association: Probabilistic ICP

- Many variations for determining the data association Δ in ICP exist:
  - data association via point-to-plane distance (Chen & Medioni, 1991)
  - probabilistic data association (EM-ICP, Granger & Pennec, 2002)

- Place a probability density function π (e.g., Gaussian) at each $\mathbf{m}_i$ to define a mixture distribution for the data:

$$p(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i \pi(\mathbf{x}; \mathbf{m}_i, \sigma_i^2 I) \qquad \alpha_i \geq 0 \qquad \sum_{i=1}^{n} \alpha_i = 1$$

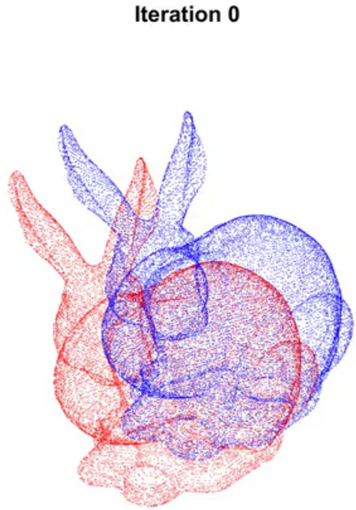
- Find parameters $\mathbf{p}$, R to maximize the likelihood of $\{R\mathbf{z}_j + \mathbf{p}\}_j$:

$$\max_{\mathbf{p},R} \sum_{j=1}^{m} \log \sum_{i=1}^{n} \alpha_i \pi(R\mathbf{z}_j + \mathbf{p}; \mathbf{m}_i, \sigma_i^2 I)$$

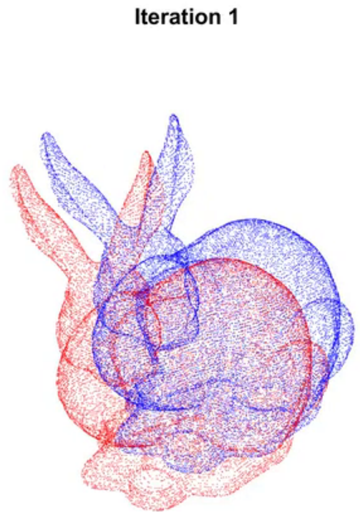
- Use **EM** to determine membership probabiliites (E step) and optimize the parameters $\mathbf{p}$, $\mathbf{R}$ (M step). ICP is a special case with $\sigma_i^2 \to 0$

- **Robustness**: use $\exp\left(-\frac{|\mathbf{x}-\mathbf{m}_i|^\beta}{2\sigma_i^2}\right)$ with $\beta \in (0,2)$ instead of $\exp\left(-\frac{|\mathbf{x}-\mathbf{m}_i|^2}{2\sigma_i^2}\right)$

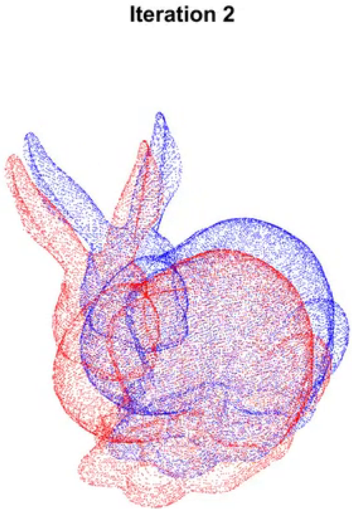
Iterative Closest Point (ICP)



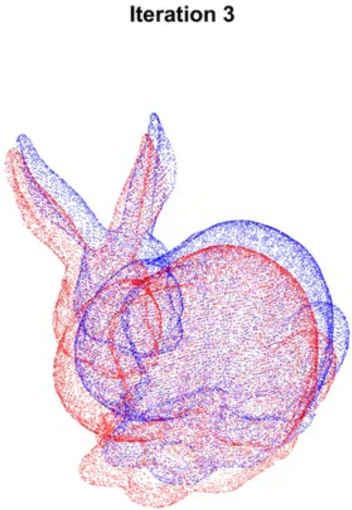
Iterative Closest Point (ICP)



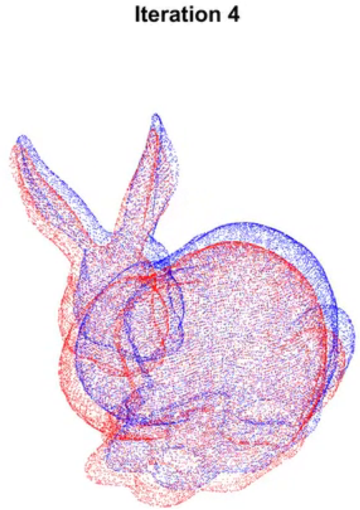
Iterative Closest Point (ICP)



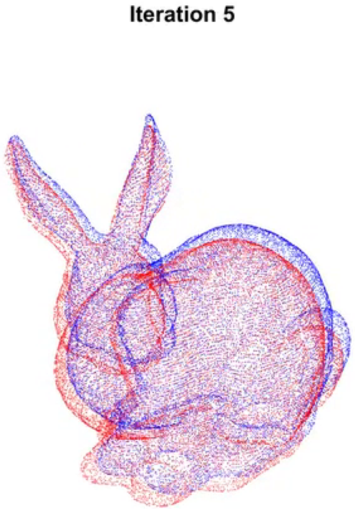
Iterative Closest Point (ICP)



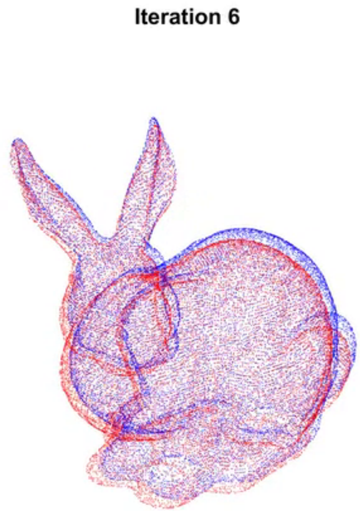
Iterative Closest Point (ICP)



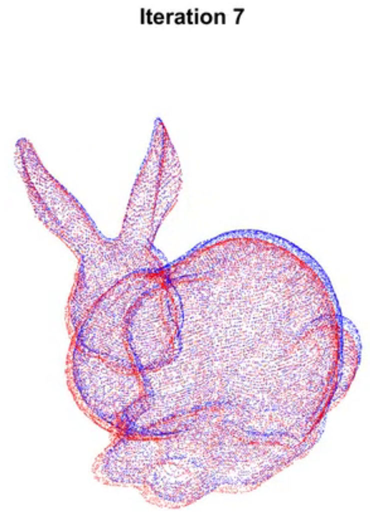
Iterative Closest Point (ICP)



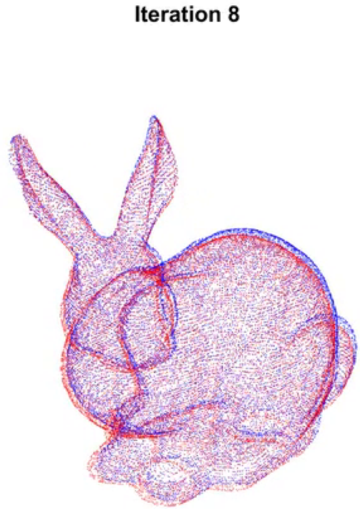
Iterative Closest Point (ICP)



Iterative Closest Point (ICP)



Iterative Closest Point (ICP)



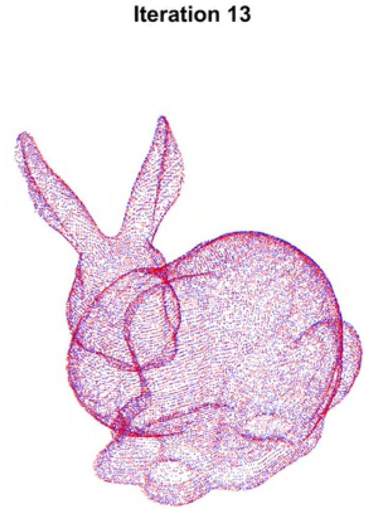
Iterative Closest Point (ICP)

Iterative Closest Point (ICP)

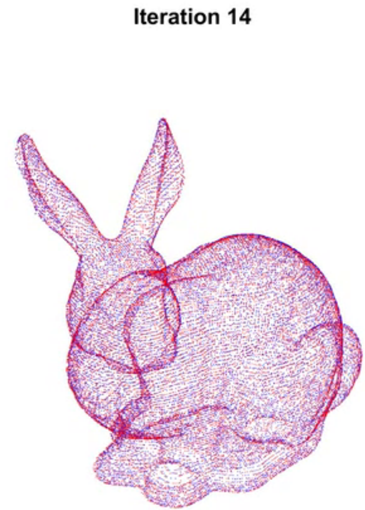
Iterative Closest Point (ICP)

Iterative Closest Point (ICP)

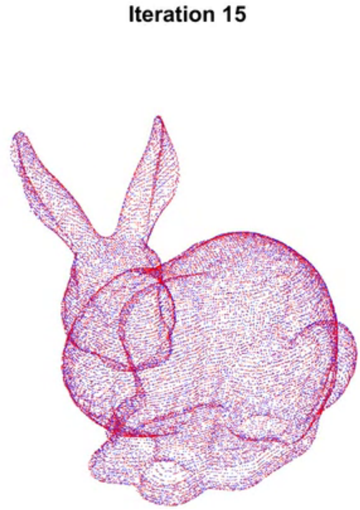
Iterative Closest Point (ICP)



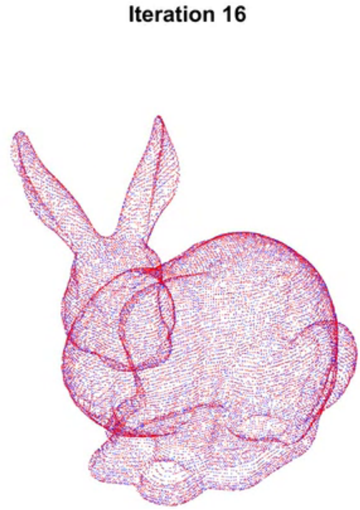
Iterative Closest Point (ICP)



Iterative Closest Point (ICP)



Iterative Closest Point (ICP)



2-D Odometry from Relative Position Measurements

- **Goal**: determine the relative transformation ${}_t\mathbf{p}_{t+1} \in \mathbb{R}^2$ and ${}_t\theta_{t+1} \in (-\pi, \pi]$ between two robot frames at time $t+1$ and t
- **Given**: relative position measurements $\mathbf{z}_{t,1}, \mathbf{z}_{t,2} \in \mathbb{R}^2$ and $\mathbf{z}_{t+1,1}, \mathbf{z}_{t+1,2} \in \mathbb{R}^2$ at consecutive time steps to two **unknown** landmarks
- If we consider the robot frame at time t to be the "world frame", this problem is **the same as 2-D localization from relative position measurements** with $\mathbf{m}_i := \mathbf{z}_{t,i}$, $\mathbf{z}_i := \mathbf{z}_{t+1,i}$, $\mathbf{p} := {}_t\mathbf{p}_{t+1}$, $\theta := {}_t\theta_{t+1}$

3-D Odometry from Relative Position Measurements

- **Goal**: determine the relative transformation ${}_t\mathbf{p}_{t+1} \in \mathbb{R}^3$ and ${}_tR_{t+1} \in SO(3)$ between two robot frames at time $t+1$ and t

- **Given**: relative position measurements $\mathbf{z}_{t,i} \in \mathbb{R}^3$ and $\mathbf{z}_{t+1,i} \in \mathbb{R}^3$ at consecutive time steps to n **unknown** landmarks

- If we consider the robot frame at time t to be the "world frame", this problem is **the same as 3-D localization from relative position measurements** with $\mathbf{m}_i := \mathbf{z}_{t,i}$, $\mathbf{z}_i := \mathbf{z}_{t+1,i}$, $\mathbf{p} := {}_t\mathbf{p}_{t+1}$, $R := {}_tR_{t+1}$

Summary: Rel. Position Measurements $\mathbf{z}_i = R^\top(\mathbf{m}_i - \mathbf{p})$

- **Localization**

$\mathbf{m}_1, \mathbf{m}_2, \mathbf{z}_1, \mathbf{z}_2 \in \mathbb{R}^2$	$(\mathbf{m}_1 - \mathbf{m}_2) = R(\theta)(\mathbf{z}_1 - \mathbf{z}_2)$ $\mathbf{p} = \frac{1}{2}\sum_{i=1}^{2}(\mathbf{m}_i - R\mathbf{z}_i)$
$\mathbf{m}_1, \mathbf{z}_i \in \mathbb{R}^3,\ i = 1,2,3$ $\mathbf{m}_{ij} := \mathbf{m}_i - \mathbf{m}_j,\ \ \mathbf{z}_{ij} := \mathbf{z}_i - \mathbf{z}_j$	$\begin{bmatrix}\mathbf{m}_{12} & \mathbf{m}_{13} & \mathbf{m}_{12} \times \mathbf{m}_{13}\end{bmatrix} = R\begin{bmatrix}\mathbf{z}_{12} & \mathbf{z}_{13} & \mathbf{z}_{12} \times \mathbf{z}_{13}\end{bmatrix}$ $\mathbf{p} = \frac{1}{3}\sum_{i=1}^{3}(\mathbf{m}_i - R\mathbf{z}_i)$
$\mathbf{m}_i, \mathbf{z}_i \in \mathbb{R}^3,\ i = 1,\ldots,n$ $\delta\mathbf{m}_i := \mathbf{m}_i - \frac{1}{n}\sum_{j=1}^{n}\mathbf{m}_j,$ $\delta\mathbf{z}_i := \mathbf{z}_i - \frac{1}{n}\sum_{j=1}^{n}\mathbf{z}_j$	$R = \arg\max_{R \in SO(3)} \sum_{i=1}^{n} \delta\mathbf{m}_i^\top R \delta\mathbf{z}_i$ $\xlongequal[SVD\left(\sum_{i=1}^{n}\delta\mathbf{m}_i\delta\mathbf{z}_i^\top\right)=U\Sigma V^\top]{\text{Kabsch algorithm}} U\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & \det(UV^\top)\end{bmatrix}V^\top$ $\mathbf{p} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{m}_i - R\mathbf{z}_i)$

- **Odometry**: same with $\mathbf{m}_i = \mathbf{z}_{t,i}$, $\mathbf{z}_i := \mathbf{z}_{t+1,i}$, $\mathbf{p} := {}_t\mathbf{p}_{t+1}$, $R := {}_tR_{t+1}$

Outline

2-D Localization from Bearing Measurements

- **Goal**: determine the robot position $\mathbf{p} \in \mathbb{R}^2$ and orientation $\theta \in (-\pi, \pi]$
- **Given**: two landmark positions $\mathbf{m}_1, \mathbf{m}_2 \in \mathbb{R}^2$ (world frame) and **bearing** measurements (body frame):

$$z_i = \arctan\left(\frac{m_{i,y} - p_y}{m_{i,x} - p_x}\right) - \theta \in \mathbb{R}, \qquad i = 1, 2$$

- The bearing constraints are equivalent to:

$$\frac{\mathbf{m}_i - \mathbf{p}}{\|\mathbf{m}_i - \mathbf{p}\|_2} = \begin{bmatrix}\cos(z_i + \theta) \\ \sin(z_i + \theta)\end{bmatrix} = R(z_i + \theta)\mathbf{e}_1 \quad \Rightarrow \quad R^\top(z_i)(\mathbf{m}_i - \mathbf{p}) = \|\mathbf{m}_i - \mathbf{p}\|_2 \begin{bmatrix}\cos(\theta) \\ \sin(\theta)\end{bmatrix}$$

- To eliminate θ, the two constraints can be combined via:

$$\begin{aligned} 0 &= \|\mathbf{m}_1 - \mathbf{p}\|_2 \begin{bmatrix}\sin\theta & -\cos\theta\end{bmatrix} \begin{bmatrix}\cos(\theta) \\ \sin(\theta)\end{bmatrix} \|\mathbf{m}_2 - \mathbf{p}\|_2 \\ &= \|\mathbf{m}_1 - \mathbf{p}\|_2 \begin{bmatrix}\cos(\theta) \\ \sin(\theta)\end{bmatrix}^\top R\left(\frac{\pi}{2}\right) \begin{bmatrix}\cos(\theta) \\ \sin(\theta)\end{bmatrix} \|\mathbf{m}_2 - \mathbf{p}\|_2 \end{aligned}$$

2-D Localization from Bearing Measurements

- The previous equation is quadratic in $\mathbf{p}$:

$$(\mathbf{m}_1 - \mathbf{p})^\top R(z_1) R\left(\frac{\pi}{2}\right) R^\top(z_2)(\mathbf{m}_2 - \mathbf{p}) = 0$$

- Let $\eta := z_1 - z_2 + \pi/2$, so that:

$$\mathbf{p}^\top R(\eta)\mathbf{p} - \left(\mathbf{m}_1^\top R(\eta) + \mathbf{m}_2^\top R^\top(\eta)\right)\mathbf{p} + \mathbf{m}_1^\top R(\eta)\mathbf{m}_2 = 0$$

- Use the following to solve the quadratic equation:
  - $\mathbf{p}^\top R(\eta)\mathbf{p} = \cos(\eta)\mathbf{p}^\top\mathbf{p}$
  - $\mathbf{p}^\top\mathbf{p} + 2\mathbf{b}^\top\mathbf{p} + c = (\mathbf{p}+\mathbf{b})^\top(\mathbf{p}+\mathbf{b}) + c - \mathbf{b}^\top\mathbf{b}$

- As long as $\cos(\eta) \neq 0$, i.e., **the robot and the two landmarks are not on the same line**:

$$(\mathbf{p} - \mathbf{p}_0)^\top(\mathbf{p} - \mathbf{p}_0) = \left(\mathbf{p}_0^\top\mathbf{p}_0 - \frac{1}{\cos(\eta)}\mathbf{m}_1^\top R(\eta)\mathbf{m}_2\right) \qquad \mathbf{p}_0 := \frac{1}{2\cos(\eta)}\left(R^\top(\eta)\mathbf{m}_1 + R(\eta)\mathbf{m}_2\right)$$

- The position $\mathbf{p}$ lies on one of the two circles containing $\mathbf{m}_1$ and $\mathbf{m}_2$

2-D Localization from Bearing Measurements

- **Pose disambiguation**: obtain a third bearing measurement:

$$R^\top(z_i)(\mathbf{m}_i - \mathbf{p}) = \|\mathbf{m}_i - \mathbf{p}\|_2 \begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix}, \quad i = 1,2,3$$

- Find β and γ such that $R^\top(z_1) + \beta R^\top(z_2) + \gamma R^\top(z_3) = 0$. Then:

$$\underbrace{R^\top(z_1)\mathbf{m}_1 + \beta R^\top(z_2)\mathbf{m}_2 + \gamma R^\top(z_3)\mathbf{m}_3}_{:=\mathbf{u}} - \underbrace{\left[R^\top(z_1) + \beta R^\top(z_2) + \gamma R^\top(z_3)\right]}_{0}\mathbf{p}$$

$$= \left(\|\mathbf{m}_1 - \mathbf{p}\|_2 + \beta\|\mathbf{m}_2 - \mathbf{p}\|_2 + \gamma\|\mathbf{m}_3 - \mathbf{p}\|_2\right)\begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix}$$

- We can compute θ as $\begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix} = \frac{\mathbf{u}}{\|\mathbf{u}\|_2}$ and recover $\mathbf{p}$ from:

$$R^\top(z_i)(\mathbf{m}_i - \mathbf{p}) = \|\mathbf{m}_i - \mathbf{p}\|_2 \begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix}, \quad i = 1,2,3$$

3-D Localization from Bearing Measurements (P3P)

- **Goal**: determine the robot position $\mathbf{p} \in \mathbb{R}^3$ and orientation $R \in SO(3)$
- **Given**: three landmark positions $\mathbf{m}_i \in \mathbb{R}^3$ (world frame) and pixel measurements $\underline{\mathbf{z}}_i \in \mathbb{R}^3$ (homogeneous coordinates, body frame) obtained from a (calibrated pinhole) camera:

$$\underline{\mathbf{z}}_i = \frac{1}{\lambda_i} R^\top (\mathbf{m}_i - \mathbf{p}) \qquad \lambda_i = \mathbf{e}_3^\top \left(R^\top (\mathbf{m}_i - \mathbf{p}) \right) = \text{unknown scale}$$

- If we determine λ_i, we can transform the P3P problem to 3-D localization from relative position measurements

Find the depths λ_i via Grunert's method

- Normalize the bearing equations:

$$\mathbf{b}_i = \frac{\underline{\mathbf{z}}_i}{\|\underline{\mathbf{z}}_i\|_2} = \frac{\lambda_i}{\lambda_i \|R^\top(\mathbf{m}_i - \mathbf{p})\|_2} R^\top(\mathbf{m}_i - \mathbf{p}) = \frac{1}{\bar{\lambda}_i} R^\top(\mathbf{m}_i - \mathbf{p})$$

where $\bar{\lambda}_i = \|R^\top(\mathbf{m}_i - \mathbf{p})\|_2 = \|\mathbf{m}_i - \mathbf{p}\|_2$

- Cosines of the angles among the bearing vectors $\mathbf{b}_1$, $\mathbf{b}_2$, $\mathbf{b}_3$:

$$\cos(\gamma_{ij}) = \frac{\mathbf{b}_i^\top \mathbf{b}_j}{\|\mathbf{b}_i\|_2 \|\mathbf{b}_j\|_2} = \mathbf{b}_i^\top \mathbf{b}_j$$

- Let $\epsilon_{ij} := \|\mathbf{m}_i - \mathbf{m}_j\|_2$ be the lengths of the triangle formed in the world frame by $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$. Applying the law of cosines gives:

$$\bar{\lambda}_i^2 + \bar{\lambda}_j^2 - 2\bar{\lambda}_i \bar{\lambda}_j \cos(\gamma_{ij}) = \epsilon_{ij}^2$$

- Let $\bar{\lambda}_2 = u\bar{\lambda}_1$ and $\bar{\lambda}_3 = v\bar{\lambda}_1$ so that:

$$\begin{aligned}
\bar{\lambda}_1^2(u^2 + v^2 - 2uv\cos(\gamma_{23})) &= \epsilon_{23}^2 \\
\bar{\lambda}_1^2(1 + v^2 - 2v\cos(\gamma_{13})) &= \epsilon_{13}^2 \\
\bar{\lambda}_1^2(u^2 + 1 - 2u\cos(\gamma_{12})) &= \epsilon_{12}^2
\end{aligned}$$

Find the depths λ_i via Grunert's method

- Equivalently

$$\bar{\lambda}_1^2 = \frac{\epsilon_{23}^2}{u^2 + v^2 - 2uv\cos(\gamma_{23})} = \frac{\epsilon_{13}^2}{1 + v^2 - 2v\cos(\gamma_{13})} = \frac{\epsilon_{12}^2}{u^2 + 1 - 2u\cos(\gamma_{12})}$$

- Cross-multiplying the second fraction, with the first and the third:

$$u^2 + \frac{\epsilon_{13}^2 - \epsilon_{23}^2}{\epsilon_{13}^2} v^2 - 2uv\cos(\gamma_{23}) + \frac{2\epsilon_{23}^2}{\epsilon_{13}^2} v\cos(\gamma_{13}) - \frac{\epsilon_{23}^2}{\epsilon_{13}^2} = 0 \qquad (1)$$

$$u^2 - \frac{\epsilon_{12}^2}{\epsilon_{13}^2} v^2 + 2v\frac{\epsilon_{12}^2}{\epsilon_{13}^2}\cos(\gamma_{13}) - 2u\cos(\gamma_{12}) + \frac{\epsilon_{13}^2 - \epsilon_{12}^2}{\epsilon_{13}^2} = 0 \qquad (2)$$

- Substituting (1) into (2):

$$u = \frac{\left(-1 + \frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right) v^2 - 2\left(\frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right)\cos(\gamma_{13})v + 1 + \frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}}{2(\cos(\gamma_{12}) - v\cos(\gamma_{23}))} \qquad (3)$$

- Substituting (3) into (1), we get a fourth-order polynomial in v:

$$a_4 v^4 + a_3 v^3 + a_2 v^2 + a_1 v + a_0 = 0$$

Polynomial Coefficients

$$a_4 = \left(\frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2} - 1\right)^2 - 4\frac{\epsilon_{12}^2}{\epsilon_{13}^2}\cos^2(\gamma_{23})$$

$$a_3 = 4\left(\frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\left(1 - \frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right)\cos(\gamma_{13}) - \left(1 - \frac{\epsilon_{23}^2 + \epsilon_{12}^2}{\epsilon_{13}^2}\right)\cos(\gamma_{23})\cos(\gamma_{12}) + 2\frac{\epsilon_{12}^2}{\epsilon_{13}^2}\cos^2(\gamma_{23})\cos(\gamma_{13})\right)$$

$$a_2 = 2\left(\left(\frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right)^2 - 1 + 2\left(\frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right)^2\cos^2(\gamma_{13}) + 2\left(\frac{\epsilon_{13}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right)\cos^2(\gamma_{23}) + 2\left(\frac{\epsilon_{13}^2 - \epsilon_{23}^2}{\epsilon_{13}^2}\right)\cos^2(\gamma_{12})\right.$$
$$\left. -4\left(\frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right)\cos(\gamma_{23})\cos(\gamma_{13})\cos(\gamma_{12})\right)$$

$$a_1 = 4\left(-\left(\frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right)\left(1 + \frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right)\cos(\gamma_{13}) - \left(1 - \frac{\epsilon_{23}^2 + \epsilon_{12}^2}{\epsilon_{13}^2}\right)\cos(\gamma_{23})\cos(\gamma_{12}) + 2\frac{\epsilon_{23}^2}{\epsilon_{13}^2}\cos^2(\gamma_{12})\cos(\gamma_{13})\right)$$

$$a_0 = \left(1 + \frac{\epsilon_{23}^2 - \epsilon_{12}^2}{\epsilon_{13}^2}\right)^2 - \frac{4\epsilon_{23}^2}{\epsilon_{13}^2}\cos^2(\gamma_{12})$$

- We can obtain up to 4 real solutions for v, which we can substitute in (3) to obtain u.
- We can recover $\bar{\lambda}_1$ from u and v via the fraction relationship
- Having $\bar{\lambda}_1$, $\bar{\lambda}_2 := u\bar{\lambda}_1$, and $\bar{\lambda}_3 := v\bar{\lambda}_1$ we have converted the P3P problem into 3-D localization from relative position measurements

3-D Localization from Bearing Measurements (PnP)

- **Goal**: determine the robot position $\mathbf{p} \in \mathbb{R}^3$ and orientation $R \in SO(3)$
- **Given**: landmark positions $\mathbf{m}_i \in \mathbb{R}^3$ (world frame) and pixel measurements $\underline{\mathbf{z}}_i \in \mathbb{R}^3$ (homogeneous coordinates) obtained from a (calibrated pinhole) camera for $i = 1, \ldots, n$:

$$\underline{\mathbf{z}}_i = \frac{1}{\lambda_i} R^\top (\mathbf{m}_i - \mathbf{p}) \qquad \lambda_i = \mathbf{e}_3^\top (R^\top (\mathbf{m}_i - \mathbf{p})) = \text{unknown scale}$$

- The PnP problem is a **constrained nonlinear least-squares** minimization:

$$\min_{\lambda_i, R, \mathbf{p}} \sum_{i=1}^{n} \left\| \underline{\mathbf{z}}_i - \frac{1}{\lambda_i} R^\top (\mathbf{m}_i - \mathbf{p}) \right\|_2^2$$

$$\text{s.t. } R^\top R = I, \quad \det R = 1, \quad \lambda_i = \mathbf{e}_3^\top (R^\top (\mathbf{m}_i - \mathbf{p}))$$

Solving the PnP Problem

- Terzakis and Lourakis, ECCV'20:
  - Eliminate the auxiliary variables λ_i and directly optimize over $\mathbf{p}$ and R
  - The optimal translation is a function of R and can be eliminated to obtain optimization in R only
  - Sequential quadratic programming with careful initialization on the 8-sphere
- Hesch and Roumeliotis, ICCV'11:
  - Express $\mathbf{p}$ and λ_i in terms of R and eliminate them to obtain an optimization in R only
  - Use Cayley-Gibbs-Rodrigues rotation parameterization to obtain a polynomial system of equations

$$R = (I + \hat{\mathbf{g}})^{-1}(I - \hat{\mathbf{g}}) = \frac{1}{1 + \mathbf{g}^\top \mathbf{g}}((1 - \mathbf{g}^\top \mathbf{g})I + 2\mathbf{g}\mathbf{g}^\top - 2\hat{\mathbf{g}})$$

where $\mathbf{g} \in \mathbb{R}^3$ is related to the angle θ and axis $\boldsymbol{\eta}$ of rotation as: $\mathbf{g} = \boldsymbol{\eta} \tan \frac{\theta}{2}$

Solving the PnP Problem (Terzakis and Lourakis, ECCV'20)

- Re-write the PnP objective in quadratic form:

$$\min_{\mathbf{r},\mathbf{b}} \sum_{i=1}^{n} (A_i\mathbf{r} + \mathbf{b})^\top Q_i (A_i\mathbf{r} + \mathbf{b})$$

  where $A_i := I \otimes \mathbf{m}_i^\top \in \mathbb{R}^{3\times 9}$, $\mathbf{r} = \text{vec}(R^\top)$, $\mathbf{b} = -R^\top\mathbf{p}$,
  $Q_i = (\underline{\mathbf{z}}_i\mathbf{e}_3^\top - I)^\top(\underline{\mathbf{z}}_i\mathbf{e}_3^\top - I) \in \mathbb{R}^{3\times 3}$

- The optimal translation is:

$$\mathbf{b} = P\mathbf{r} \qquad P = -\left(\sum_{i=1}^{n} Q_i\right)^{-1}\left(\sum_{i=1}^{n} Q_i A_i\right)$$

- With $\Omega = \sum_{i=1}^{n}(A_i + P)^\top Q_i (A_i + P)$, we get a non-linear quadratic program:

$$\min_{\text{mat}(\mathbf{r})\in SO(3)} \mathbf{r}^\top\Omega\mathbf{r}$$

- Use sequential quadratic programming initialized from solutions of $\min_{\mathbf{r}\in\mathbb{S}^8} \mathbf{r}^\top\Omega\mathbf{r}$

Solving the PnP Problem (Hesch and Roumeliotis, ICCV'11)

- The constraints $\lambda_i \underline{\mathbf{z}}_i = R^\top(\mathbf{m}_i - \mathbf{p})$ can be re-written in matrix form as:

$$\underbrace{\begin{bmatrix} \underline{\mathbf{z}}_1 & & & -I \\ & \ddots & & \vdots \\ & & \underline{\mathbf{z}}_n & -I \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \\ -R^\top \mathbf{p} \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} R^\top & & \\ & \ddots & \\ & & R^\top \end{bmatrix}}_{W} \underbrace{\begin{bmatrix} \mathbf{m}_1 \\ \vdots \\ \mathbf{m}_n \end{bmatrix}}_{\mathbf{d}}$$

  where A and $\mathbf{d}$ are known or measured, $\mathbf{x}$ are the unknowns we wish to eliminate, and W is a block diagonal matrix of the unknown rotation R

- Express $\mathbf{p}$ and λ_i in terms of the other quantities:

$$\mathbf{x} = (A^\top A)^{-1} A^\top W \mathbf{d} = \begin{bmatrix} U \\ V \end{bmatrix} W \mathbf{d}$$

  where $(A^\top A)^{-1} A^\top$ is partitioned so that the scale parameters are a function of U and the translation $-R^\top \mathbf{p}$ is a function of V.

Solving the PnP Problem (Hesch and Roumeliotis, ICCV'11)

$$\mathbf{x} = (A^\top A)^{-1} A^\top W \mathbf{d} = \begin{bmatrix} U \\ V \end{bmatrix} W \mathbf{d}$$

- Exploiting the sparse structure of A, the matrices U and V can be computed in closed form
- Both λ_i and $-R^\top \mathbf{p}$ are linear functions of the unknown $R^\top$:

  $$\lambda_i = \mathbf{u}_i^\top W \mathbf{d} \qquad - R^\top \mathbf{p} = V W \mathbf{d}, \qquad i = 1, \ldots, n$$

  where $\mathbf{u}_i^\top$ is the i-th row of U
- We can rewrite the constraints $\lambda_i \underline{\mathbf{z}}_i = R^\top(\mathbf{m}_i - \mathbf{p})$ as:

  $$\underbrace{\mathbf{u}_i^\top W \mathbf{d}}_{\lambda_i} \underline{\mathbf{z}}_i = R^\top \mathbf{m}_i + \underbrace{V W \mathbf{d}}_{-R^\top \mathbf{p}}$$
- We have reduced the number of unknowns from $6 + n$ to 3

Solving the PnP Problem (Hesch and Roumeliotis, ICCV'11)

- **Cayley-Gibbs-Rodrigues rotation parameterization**:

$$R^\top = \frac{\bar{C}}{1+\mathbf{g}^\top\mathbf{g}} \qquad \bar{C} = (I-\hat{\mathbf{g}})^{-1}(I+\hat{\mathbf{g}}) = ((1-\mathbf{g}^\top\mathbf{g})I_3 + 2\hat{\mathbf{g}} + 2\mathbf{g}\mathbf{g}^\top)$$

- The CGR parameters automatically satisfy the rotation matrix constraints, i.e., $R^\top R = I$ and $\det(R) = 1$ and allow us to formulate an unconstrained least-squares minimization in $\mathbf{g}$.

- **Reformulation into a polynomial system**: Since $R^\top$ appears linearly in the equations, we can cancel the denominator $1+\mathbf{g}^\top\mathbf{g}$. This leads to the following formulation of the PnP problem:

$$\min_{\mathbf{g}} J(\mathbf{g}) = \sum_{i=1}^{n} \left\| \mathbf{u}_i^\top \begin{bmatrix} \bar{C} & & \\ & \ddots & \\ & & \bar{C} \end{bmatrix} \mathbf{d}\underline{\mathbf{z}}_i - \bar{C}\mathbf{m}_i - V \begin{bmatrix} \bar{C} & & \\ & \ddots & \\ & & \bar{C} \end{bmatrix} \mathbf{d} \right\|^2$$

which contains all monomials up to degree four, i.e.,
$\{1, g_1, g_2, g_3, g_1g_2, g_1g_3, g_2g_3, \ldots, g_1^4, g_2^4, g_3^4\}$.

Solving the PnP Problem (Hesch and Roumeliotis, ICCV'11)

- Since $J(\mathbf{g})$ is a fourth-order polynomial, the optimality conditions form a system of three third-order polynomials (derivatives with respect to g_1, g_2 and g_3).
- Use a **Macaulay resultant matrix** (matrix of polynomial coefficients) to find the roots of the third-order polynomials and hence compute all critical points of $J(\mathbf{g})$
- Since the polynomial system is of constant degree (independent of n), it is only necessary to compute the Macaulay matrix symbolically once
- Online, the elements of the Macaulay matrix are formed from the data (linear operation in n) and the roots are determined via an eigen-decomposition of the Schur complement (dense 27×27 matrix) of the top block of the Macaulay matrix (sparse 120×120 matrix)

2-D Odometry from Bearing Measurements

- **Goal**: determine the relative transformation ${}_t\mathbf{p}_{t+1} \in \mathbb{R}^2$ and ${}_t\theta_{t+1} \in (-\pi, \pi]$ between two robot frames at time $t+1$ and t
- **Given**: bearing measurements $\mathbf{z}_{t,i} \in \mathbb{R}^2$ and $\mathbf{z}_{t+1,i} \in \mathbb{R}^2$ (unit vectors) at consecutive time steps to n **unknown** landmarks
- The measurements are related as follows:

$$d_{t,i}\mathbf{b}_{t,i} = {}_t\mathbf{p}_{t+1} + d_{t+1,i}R({}_t\theta_{t+1})\mathbf{b}_{t+1,i}, \qquad i = 1, \ldots, n$$

  where $d_{t,i}, d_{t+1,i}$ are the unknown distances to $\mathbf{m}_i$.
- There are $2n$ equations and $2n+3$ unknowns, which means that this problem is **not solvable**.

3-D Odometry from Bearing Measurements

- **Goal**: determine the relative transformation ${}_t\mathbf{p}_{t+1} \in \mathbb{R}^3$ and ${}_tR_{t+1} \in SO(3)$ between two robot frames at time $t+1$ and t
- **Given**: pixel coordinates $\underline{\mathbf{z}}_{t,i} \in \mathbb{R}^3$ and $\underline{\mathbf{z}}_{t+1,i} \in \mathbb{R}^3$ at consecutive time steps to n **unknown** landmarks ($n \geq 5$) with **known** camera calibration matrices K_t and K_{t+1}
- Without loss of generality, assume that the first camera frame coincides with the world frame and denote $\mathbf{p} = {}_t\mathbf{p}_{t+1}$ and $R = {}_tR_{t+1}$
- Let $\underline{\mathbf{y}}_{t,i} := K_t^{-1}\underline{\mathbf{z}}_{t,i}$ and $\underline{\mathbf{y}}_{t+1,i} := K_{t+1}^{-1}\underline{\mathbf{z}}_{t+1,i}$ be the normalized pixel coordinates so that:

$$\lambda_{t,i}\underline{\mathbf{y}}_{t,i} = \mathbf{m}_i, \qquad \lambda_{t,i} = \mathbf{e}_3^\top \mathbf{m}_i = \text{unknown depth}$$

$$\lambda_{t+1,i}\underline{\mathbf{y}}_{t+1,i} = R^\top(\mathbf{m}_i - \mathbf{p}), \qquad \lambda_{t+1,i} = \mathbf{e}_3^\top R^\top(\mathbf{m}_i - \mathbf{p}) = \text{unknown depth}$$

Epipolar Constraint and Essential Matrix

- The pixel projections of landmark $\mathbf{m}_i$ in the two images satisfy:

$$\lambda_{t,i}\underline{\mathbf{y}}_{t,i} = \lambda_{t+1,i}R\underline{\mathbf{y}}_{t+1,i} + \mathbf{p}$$

- To eliminate the unknown depths $\lambda_{t,i}$, $\lambda_{t+1,i}$, pre-multiply by $\hat{\mathbf{p}}$ and note that $\hat{\mathbf{p}}\underline{\mathbf{y}}_{t,i}$ is perpendicular to $\underline{\mathbf{y}}_{t,i}$:

$$\underbrace{\lambda_{t,i}\underline{\mathbf{y}}_{t,i}^\top\hat{\mathbf{p}}\underline{\mathbf{y}}_{t,i}}_{0} = \lambda_{t+1,i}\underline{\mathbf{y}}_{t,i}^\top\hat{\mathbf{p}}R\underline{\mathbf{y}}_{t+1,i} + \underbrace{\underline{\mathbf{y}}_{t,i}^\top\hat{\mathbf{p}}\mathbf{p}}_{0}$$

- **Epipolar constraint**: the normalized pixel coordinates $\underline{\mathbf{y}}_{t,i} = K_t^{-1}\underline{\mathbf{z}}_{t,i}$ and $\underline{\mathbf{y}}_{t+1,i} = K_{t+1}^{-1}\underline{\mathbf{z}}_{t+1,i}$ of the same point $\mathbf{m}_i$ in two calibrated cameras with relative pose $(R, \mathbf{p})$ of cam 2 in the frame of cam 1 satisfy:

$$0 = \underline{\mathbf{y}}_{t,i}^\top\left(\hat{\mathbf{p}}R\right)\underline{\mathbf{y}}_{t+1,i} = \underline{\mathbf{y}}_{t,i}^\top E\underline{\mathbf{y}}_{t+1,i}$$

  where $E := \hat{\mathbf{p}}R \in \mathbb{R}^{3\times 3}$ is the **essential matrix**

- **Essential matrix characterization**: a non-zero $E \in \mathbb{R}^{3\times 3}$ is an essential matrix iff its singular value decomposition is $E = U\mathbf{diag}(\sigma, \sigma, 0)V^\top$ for some $\sigma \geq 0$ and $U, V \in SO(3)$

3-D Odometry from Bearing Measurements (8-Pt Alg)

- The epipolar constraint $0 = \underline{\mathbf{y}}_{t,i}^\top E \underline{\mathbf{y}}_{t+1,i}$ is linear in the elements of E:

$$0 = \bar{\mathbf{y}}_i^\top \mathbf{e}$$

  where $\bar{\mathbf{y}}_i := \text{vec}(\underline{\mathbf{y}}_{t,i}\underline{\mathbf{y}}_{t+1,i}^\top) \in \mathbb{R}^9$, $\mathbf{e} := \text{vec}(E) \in \mathbb{R}^9$, and $\text{vec}(\cdot)$ is the vectorization of a matrix, which stacks its columns into a vector

- Stacking $\bar{\mathbf{y}}_i$ from all 8 observations together, we obtain an 8×9 matrix $\bar{Y} := \begin{bmatrix} \bar{\mathbf{y}}_1 & \cdots & \bar{\mathbf{y}}_8 \end{bmatrix}^\top$ leading to the following equation for $\mathbf{e}$:

$$\bar{Y}\mathbf{e} = 0$$

- Thus, $\mathbf{e}$ is a singular vector of $\bar{Y}$ associated to a singular value that equals zero

- If at least 8 linearly independent vectors $\bar{\mathbf{y}}_i$ are used to construct $\bar{Y}$, then the singular vector is unique (up to scalar multiplication) and $\mathbf{e}$ and E can be determined

3-D Odometry from Bearing Measurements (5-Pt Alg)

- The essential matrix E can be recovered from $\bar{Y}\mathbf{e} = 0$, even if only 5 linearly independent vectors $\bar{\mathbf{y}}_i$ are available using the fact that:

$$0 = EE^\top E - \frac{1}{2}\operatorname{tr}(EE^\top)E$$

- Stacking $\bar{\mathbf{y}}_i$'s together, we obtain a 5×9 matrix $\bar{Y} := \begin{bmatrix}\bar{\mathbf{y}}_1 & \cdots & \bar{\mathbf{y}}_5\end{bmatrix}^\top$

- The right nullspace of $\bar{Y}$ has dimension 4 and the vectors that span the nullspace (obtained from SVD or QR decomposition) correspond to 3×3 matrices N_i, $i = 1, \ldots, 4$ such that

$$E = \alpha_1 N_1 + \alpha_2 N_2 + \alpha_3 N_3 + \alpha_4 N_4, \qquad \alpha_i \in \mathbb{R}$$

- Since the measurements are scale-invariant, we can arbitrarily fix $\alpha_4 = 1$

- Substituting $E = \alpha_1 N_1 + \alpha_2 N_2 + \alpha_3 N_3 + N_4$, we obtain 9 cubic-in-$\alpha_i$ equations and can recover up to 10 solutions for E

3-D Odometry from Bearing Measurements

- Once E is recovered, $\mathbf{p}$ and R can be computed from the singular value decomposition of E
- **Pose recovery from the essential matrix**: there are exactly two relative poses corresponding to a non-zero essential matrix $E = U\mathbf{diag}(\sigma, \sigma, 0)V^\top$:

$$(\hat{\mathbf{p}}, R) = \left(UR_z\left(\frac{\pi}{2}\right)\mathbf{diag}(\sigma,\sigma,0)U^\top, UR_z^\top\left(\frac{\pi}{2}\right)V^\top \right)$$
$$(\hat{\mathbf{p}}, R) = \left(UR_z\left(-\frac{\pi}{2}\right)\mathbf{diag}(\sigma,\sigma,0)U^\top, UR_z^\top\left(-\frac{\pi}{2}\right)V^\top \right)$$

- Only one of these will place the points in front of both cameras
- The ambiguity can be resolved by intersecting the measurements of a single point and verifying which solution places it on the positive optical z-axis of both cameras

Bearing Measurement Triangulation

- **Goal**: determine the coordinates of a point $\mathbf{m} \in \mathbb{R}^3$ observed by two cameras in the reference frame of the first camera

- **Given**: pixel coordinates $\mathbf{z}_1 \in \mathbb{R}^2$ and $\mathbf{z}_2 \in \mathbb{R}^2$ obtained from two calibrated cameras with known relative transformation $\mathbf{p} \in \mathbb{R}^3$ and $R \in SO(3)$ of cam 2 in the frame of cam 1:

$$\lambda_1 \underline{\mathbf{z}}_1 = \mathbf{m}, \qquad \lambda_1 = \mathbf{e}_3^\top \mathbf{m} = \text{unknown depth}$$

$$\lambda_2 \underline{\mathbf{z}}_2 = R^\top (\mathbf{m} - \mathbf{p}), \qquad \lambda_2 = \mathbf{e}_3^\top R^\top (\mathbf{m} - \mathbf{p}) = \text{unknown depth}$$

- We can determine $\mathbf{m} = \lambda_1 \underline{\mathbf{z}}_1$ by solving for the unknown depth λ_1 using the second measurement equation

- Note that $\lambda_2 = \lambda_1 \mathbf{e}_3^\top \mathbf{R}^\top \underline{\mathbf{z}}_1 - \mathbf{e}_3^\top \mathbf{R}^\top \mathbf{p}$ and thus:

$$\left(\lambda_1 \mathbf{e}_3^\top \mathbf{R}^\top \underline{\mathbf{z}}_1 - \mathbf{e}_3^\top \mathbf{R}^\top \mathbf{p}\right) \underline{\mathbf{z}}_2 = \lambda_1 \mathbf{R}^\top \underline{\mathbf{z}}_1 - \mathbf{R}^\top \mathbf{p}$$

$$\underbrace{\left(\mathbf{R}^\top \mathbf{p} - \mathbf{e}_3^\top \mathbf{R}^\top \mathbf{p} \underline{\mathbf{z}}_2\right)}_{\mathbf{a}} \frac{1}{\lambda_1} = \underbrace{\left(\mathbf{R}^\top \underline{\mathbf{z}}_1 - \mathbf{e}_3^\top \mathbf{R}^\top \underline{\mathbf{z}}_1 \underline{\mathbf{z}}_2\right)}_{\mathbf{b}}$$

$$\frac{1}{\lambda_1} = \frac{\mathbf{a}^\top \mathbf{b}}{\mathbf{a}^\top \mathbf{a}} \qquad \Rightarrow \qquad \mathbf{m} = \frac{\mathbf{a}^\top \mathbf{a}}{\mathbf{a}^\top \mathbf{b}} \underline{\mathbf{z}}_1$$

Summary: Bearing Measurements $\underline{\mathbf{z}}_i = \frac{1}{\lambda_i} R^\top(\mathbf{m}_i - \mathbf{p})$

- **2-D Localization**: given $\mathbf{m}_1, \mathbf{m}_2 \in \mathbb{R}^2$ and $z_1, z_2 \in [-\pi, \pi]$
  1. 2-D bearing: $\frac{1}{\lambda_i} R^\top(\theta)(\mathbf{m}_i - \mathbf{p}) = R(z_i)\mathbf{e}_1$
  2. Eliminate θ:

$$0 = \lambda_1 \mathbf{e}_1^\top R(\theta) R\left(\frac{\pi}{2}\right) R(\theta)\mathbf{e}_1 \lambda_2 = (\mathbf{m}_1 - \mathbf{p})^\top R(z_1) R\left(\frac{\pi}{2}\right) R^\top(z_2)(\mathbf{m}_2 - \mathbf{p})$$

  3. The position $\mathbf{p}$ in on one of two circles containing $\mathbf{m}_1$ and $\mathbf{m}_2$ and we need a third bearing measurement z_3 to disambiguate it
  4. Find β, γ such that $R^\top(z_1) + \beta R^\top(z_2) + \gamma R^\top(z_3) = 0$ and combine $R^\top(z_i)(\mathbf{m}_i - \mathbf{p}) = \lambda_i \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix}$ to solve for θ
  5. Orientation: $\begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} = \frac{\mathbf{u}}{\|\mathbf{u}\|_2}$ for $\mathbf{u} = R^\top(z_1)\mathbf{m}_1 + \beta R^\top(z_2)\mathbf{m}_2 + \gamma R^\top(z_3)\mathbf{m}_3$

- **3-D Localization (P3P)**: $\mathbf{m}_i \in \mathbb{R}^3$, $\underline{\mathbf{z}}_i \in \mathbb{R}^3$ (homogeneous), $i = 1, 2, 3$
  1. Convert P3P to relative position localization by determining the depths $\lambda_1, \lambda_2, \lambda_3$ via Grunert's method
  2. Define angles γ_{ij} among normalized $\underline{\mathbf{z}}_1, \underline{\mathbf{z}}_2, \underline{\mathbf{z}}_3$ and apply the law of cosines: $\lambda_i^2 + \lambda_j^2 - 2\lambda_i\lambda_j \cos(\gamma_{ij}) = \|\mathbf{m}_1 - \mathbf{m}_j\|_2^2$
  3. Let $\lambda_2 = u\lambda_1$ and $\lambda_3 = v\lambda_1$ and combine the 3 equations to get a fourth order polynomial: $a_4 v^4 + a_3 v^3 + a_2 v^2 + a_1 v + a_0 = 0$

Summary: Bearing Measurements $\underline{\mathbf{z}}_i = \frac{1}{\lambda_i} R^\top (\mathbf{m}_i - \mathbf{p})$

- **3-D Localization (PnP)**
  1. Rewrite $\lambda_i \underline{\mathbf{z}}_i = R^\top (\mathbf{m}_i - \mathbf{p})$ in matrix form and solve for $\mathbf{x} := (\lambda_1, \ldots, \lambda_n, -R^\top \mathbf{p})^\top$ in terms of R
  2. The equations for λ_i and $-R^\top \mathbf{p}$ turn out to be linear in R so we are left with one equation with 3 unknowns (the 3 degrees of freedom of R)
  3. Obtain a fourth order polynomial $J(\mathbf{g})$ in terms of the Cayley-Gibbs-Rodrigues rotation parameterization $\mathbf{g}$
  4. Compute a Macaulay matrix of the coefficients of $J(\mathbf{g})$ symbolically once. Online, determine the roots of $J(\mathbf{g})$ via an eigen-decomposition of the Schur complement of the Macaulay matrix.

- **2-D Odometry**: not solvable

- **3-D Odometry**: 5-point or 8-point algorithm:
  1. Obtain E from the epipolar constraint: $0 = \text{vec}\left(\underline{\mathbf{y}}_{t,i} \underline{\mathbf{y}}_{t+1,i}^\top\right)^\top \text{vec}(E)$, $i = 1, \ldots, 5$ and the property $0 = EE^\top E - \frac{1}{2}\text{tr}(EE^\top)E$
  2. Recover two possible camera poses based on $SVD(E) = U\mathbf{diag}(\sigma, \sigma, 0)V^\top$ and choose the one that places the measurements in front of both cameras

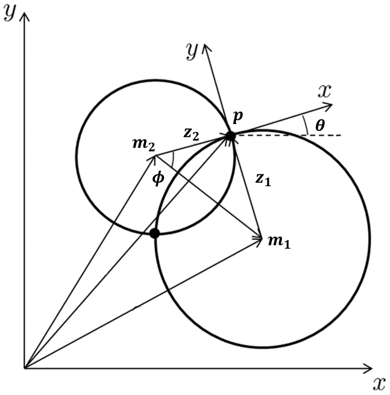
Outline

2-D Localization from Range Measurements

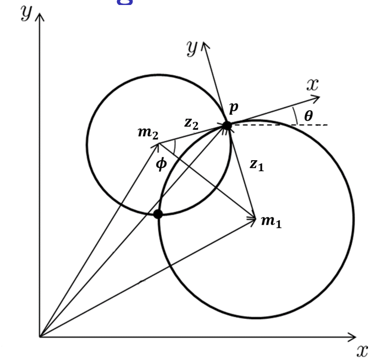
- **Goal**: determine the robot position $\mathbf{p} \in \mathbb{R}^2$ and orientation $\theta \in (-\pi, \pi]$
- **Given**: two landmark positions $\mathbf{m}_1, \mathbf{m}_2 \in \mathbb{R}^2$ (world frame) and **range** measurements (body frame):

$$z_i = \|\mathbf{m}_i - \mathbf{p}\|_2 \in \mathbb{R}, \quad i = 1, 2$$

- Because all possible positions whose distance to $\mathbf{m}_1$ is z_1 is a circle, the possible robot positions are given by the intersection of two circles



2-D Localization from Range Measurements



- Applying the law of cosines to the triangle gives:

$$z_2^2 = z_1^2 + \|\mathbf{m}_2 - \mathbf{m}_1\|_2^2 - 2z_1\|\mathbf{m}_2 - \mathbf{m}_1\|_2 \cos\phi$$

- Solving for ϕ and then the circle intersection points provides the possible robot positions:

$$\mathbf{p} = \mathbf{m}_2 + z_2 R(\pm\phi)\frac{\mathbf{m}_1 - \mathbf{m}_2}{\|\mathbf{m}_1 - \mathbf{m}_2\|_2}$$

- The orientation of the robot θ is **not identifiable**

2-D Localization from Range Measurements

- **Pose disambiguation**: the robot can make a move with known translation $\mathbf{p}_\Delta$ (measured in the frame at time t) and take two new range measurements
- There are 2 possible robot positions at each time frame for a total of 4 combinations but comparing $\|\mathbf{p}_{t+1} - \mathbf{p}_t\|_2$ to the known $\|\mathbf{p}_\Delta\|_2$ leaves only two valid options (and we cannot distinguish between them)
- To obtain the orientation, we use geometric constraints:

$$\mathbf{p}_{t+1} - \mathbf{p}_t = R(\theta_t)\mathbf{p}_\Delta = \begin{bmatrix} p_{\Delta,x} & -p_{\Delta,y} \\ p_{\Delta,y} & p_{\Delta,x} \end{bmatrix} \begin{bmatrix} \cos\theta_t \\ \sin\theta_t \end{bmatrix}$$

- As long as $\det \begin{bmatrix} p_{\Delta,x} & -p_{\Delta,y} \\ p_{\Delta,y} & p_{\Delta,x} \end{bmatrix} = \|\mathbf{p}_\Delta\|_2^2 \neq 0$, we can compute:

$$\begin{bmatrix} \cos\theta_t \\ \sin\theta_t \end{bmatrix} = \frac{1}{\|\mathbf{p}_\Delta\|_2^2} \begin{bmatrix} p_{\Delta,x} & p_{\Delta,y} \\ -p_{\Delta,y} & p_{\Delta,x} \end{bmatrix} (\mathbf{p}_{t+1} - \mathbf{p}_t)$$

$$\theta_t = \mathbf{atan2}(\sin\theta_t, \cos\theta_t)$$

3-D Localization from Range Measurements

- **Goal**: determine the robot position $\mathbf{p} \in \mathbb{R}^3$ and orientation $R \in SO(3)$
- **Given**: three landmark positions $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3 \in \mathbb{R}^3$ (world frame) and **range** measurements (body frame):

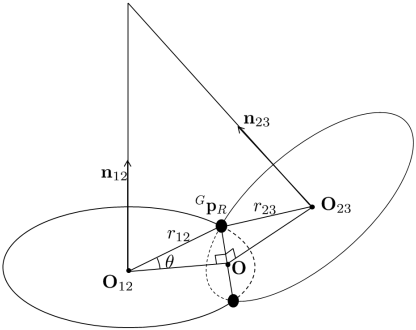
$$z_i = \|\mathbf{m}_i - \mathbf{p}\|_2 \in \mathbb{R}, \quad i = 1, 2, 3$$

- All possible positions whose distance to $\mathbf{m}_1$ is z_1 is a sphere
- The possible robot positions are the intersections of three spheres
- To find the intersection of 3 spheres, we first find the intersection of sphere one and two (a circle) and of sphere two and three (a circle). The intersection of these two circles gives the possible robot positions.
- **Degenerate case**: all landmarks are on the same line – the intersection of the spheres is a circle with infinitely many possible robot positions

3-D Localization from Range Measurements

- **Intersecting circle of spheres with radii z_1 and z_2**: center $\mathbf{o}_{12}$, radius r_{12}, normal vector $\mathbf{n}_{12}$ (perpendicular to the circle plane)
- Law of Cosines: $z_2^2 = z_1^2 + \|\mathbf{m}_2 - \mathbf{m}_1\|_2^2 - 2z_1\|\mathbf{m}_2 - \mathbf{m}_1\|_2 \cos\theta_{12}$
- Geometric relationships:

$$\mathbf{o}_{12} = \mathbf{m}_1 + z_1 \cos\theta_{12}\mathbf{n}_{12}$$
$$r_{12} = z_1|\sin(\theta_{12})|$$
$$\mathbf{n}_{12} = \frac{\mathbf{m}_2 - \mathbf{m}_1}{\|\mathbf{m}_2 - \mathbf{m}_1\|_2}$$



- **Intersecting circle of spheres with radii z_2 and z_3**: center $\mathbf{o}_{23}$, radius r_{23}, normal vector $\mathbf{n}_{23}$ (perpendicular to the circle plane):

$$\mathbf{o}_{23} = \mathbf{m}_2 + z_2 \cos\theta_{23}\mathbf{n}_{23} \qquad r_{23} = z_2|\sin(\theta_{23})| \qquad \mathbf{n}_{23} = \frac{\mathbf{m}_3 - \mathbf{m}_2}{\|\mathbf{m}_3 - \mathbf{m}_2\|_2}$$

3-D Localization from Range Measurements

- The intersecting points of the two circles can be obtained from:

$$\begin{aligned} \mathbf{n}_{12}^\top(\mathbf{o}_{12} - \mathbf{o}) &= 0 \\ \mathbf{n}_{23}^\top(\mathbf{o}_{23} - \mathbf{o}) &= 0 \\ (\mathbf{n}_{12} \times \mathbf{n}_{23})^\top(\mathbf{o}_{12} - \mathbf{o}) &= 0 \end{aligned} \qquad \begin{bmatrix} \mathbf{n}_{12}^\top \\ \mathbf{n}_{23}^\top \\ (\mathbf{n}_{12} \times \mathbf{n}_{23})^\top \end{bmatrix} \mathbf{o} = \begin{bmatrix} \mathbf{n}_{12}^\top \mathbf{o}_{12} \\ \mathbf{n}_{23}^\top \mathbf{o}_{23} \\ (\mathbf{n}_{12} \times \mathbf{n}_{23})^\top \mathbf{o}_{12} \end{bmatrix}$$

- As long as the three landmarks are not on the same line, we can uniquely solve for **o**:

$$\det \begin{bmatrix} \mathbf{n}_{12}^\top \\ \mathbf{n}_{23}^\top \\ (\mathbf{n}_{12} \times \mathbf{n}_{23})^\top \end{bmatrix} \neq 0 \qquad \Leftrightarrow \qquad \mathbf{n}_{12} \text{ and } \mathbf{n}_{23} \text{ not colinear}$$

- The two possible robot positions are:

$$\mathbf{p} = \mathbf{o}_{12} + r_{12} R(\mathbf{n}_{12}, \pm\theta) \frac{\mathbf{o} - \mathbf{o}_{12}}{\|\mathbf{o} - \mathbf{o}_{12}\|_2} \qquad \cos\theta = \frac{\|\mathbf{o} - \mathbf{o}_{12}\|_2}{r_{12}}$$

- As in the 2-D case, the robot orientation R is **not identifiable**

3-D Localization from Range Measurements

- **Pose disambiguation**: the robot can make a move with known translation $\mathbf{p}_\Delta \in \mathbb{R}^3$ and rotation $R_\Delta \in SO(3)$ and take three new range measurements
- As in the 2-D case, after eliminating the impossible robot positions, we should be left with only two options for $\mathbf{p}_t$ and $\mathbf{p}_{t+1}$
- Given $\mathbf{p}_t$, $\mathbf{p}_{t+1}$, $\mathbf{p}_\Delta$, and R_Δ, we can now obtain R_t

$$\mathbf{p}_{t+1} = \mathbf{p}_t + R_t \mathbf{p}_\Delta$$

- This is not sufficient because the rotation about $\mathbf{p}_\Delta$ is not identifiable
- The robot needs to **move a second time** to a third pose $\mathbf{p}_{t+2}, R_{t+2}$ with known translation $\mathbf{p}_{\Delta,2} \in \mathbb{R}^3$ and take three more range measurements to the three landmarks:

$$\mathbf{p}_{t+2} = \mathbf{p}_{t+1} + R_{t+1}\mathbf{p}_{\Delta,2} = \mathbf{p}_{t+1} + R_t R_\Delta \mathbf{p}_{\Delta,2}$$

3-D Localization from Range Measurements

- Putting the previous two equations together:

$$\begin{aligned}\mathbf{p}_{t+1} - \mathbf{p}_t &= R_t \mathbf{p}_\Delta \\ \mathbf{p}_{t+2} - \mathbf{p}_{t+1} &= R_t R_\Delta \mathbf{p}_{\Delta,2}\end{aligned}$$

- Taking a cross product between the two:

$$(\mathbf{p}_{t+1} - \mathbf{p}_t) \times (\mathbf{p}_{t+2} - \mathbf{p}_{t+1}) = R_t(\mathbf{p}_\Delta \times R_\Delta \mathbf{p}_{\Delta,2})$$

- As long as $U := [\mathbf{p}_\Delta,\ R_\Delta \mathbf{p}_{\Delta,2},\ \mathbf{p}_\Delta \times R_\Delta \mathbf{p}_{\Delta,2})]$ is nonsingular, i.e., $\mathbf{p}_\Delta$ and $R_\Delta \mathbf{p}_{\Delta,2}$ are not co-linear or equivalently **the three robot positions are not on the same line**, we can determine the robot orientation:

$$R_t = [(\mathbf{p}_{t+1} - \mathbf{p}_t),\ (\mathbf{p}_{t+2} - \mathbf{p}_{t+1}),\ (\mathbf{p}_{t+1} - \mathbf{p}_t) \times (\mathbf{p}_{t+2} - \mathbf{p}_{t+1})]U^{-1}$$

2-D Odometry from Range Measurements

- **Goal**: determine the relative transformation ${}_t\mathbf{p}_{t+1} \in \mathbb{R}^2$ and ${}_t\theta_{t+1} \in (-\pi, \pi]$ between two robot frames at time $t+1$ and t
- **Given**: range measurements $z_{t,i} \in \mathbb{R}$ and $z_{t+1,i} \in \mathbb{R}$ at consecutive time steps to n **unknown** landmarks
- Let $\mathbf{m}_{t+1,i}$ be the relative position to the i-th landmark at $t+1$ so that:

$$
\begin{aligned}
z_{t+1,i} &= \|\mathbf{m}_{t+1,i}\|_2 \\
z_{t,i} &= \|{}_t\mathbf{p}_{t+1} + R({}_t\theta_{t+1})\mathbf{m}_{t+1,i}\|_2
\end{aligned}
$$

- Squaring and combining these equations, we get:

$$
[{}_t\mathbf{p}_{t+1}]^\top {}_t\mathbf{p}_{t+1} + 2\mathbf{m}_{t+1,i}^\top R^\top({}_t\theta_{t+1}){}_t\mathbf{p}_{t+1} = z_{t,i}^2 - z_{t+1,i}^2, \quad i = 1, \ldots, n
$$

- We have n equations with $n+3$ unknowns (3 for the relative pose and n for the unknown directions to the landmarks at $t+1$), which is **not solvable**.

3-D Odometry from Range Measurements

- **Goal**: determine the relative transformation ${}_t\mathbf{p}_{t+1} \in \mathbb{R}^3$ and ${}_tR_{t+1} \in SO(3)$ between two robot frames at time $t+1$ and t

- **Given**: range measurements $z_{t,i} \in \mathbb{R}$ and $z_{t+1,i} \in \mathbb{R}$ at consecutive time steps to n **unknown** landmarks

- Following the same derivation as in the 2-D case, we obtain:

$$\left[{}_t\mathbf{p}_{t+1}\right]^\top {}_t\mathbf{p}_{t+1} + 2\mathbf{m}_{t+1,i}^\top \left[{}_tR_{t+1}\right]^\top {}_t\mathbf{p}_{t+1} = z_{t,i}^2 - z_{t+1,i}^2, \quad i = 1, \ldots, n$$

- We have n equations with $2n+6$ unknowns (6 for the relative pose and $2n$ for the unknown directions to the landmarks at $t+1$), which is **not solvable**.

Summary: Range Measurements $z_i = \|\mathbf{m}_i - \mathbf{p}\|_2$

- **2-D Localization**: given $\mathbf{m}_1, \mathbf{m}_2 \in \mathbb{R}^2$ and $z_1, z_2 \in \mathbb{R}$
  1. Law of Cosines: $z_2^2 = z_1^2 + \|\mathbf{m}_2 - \mathbf{m}_1\|_2^2 - 2z_1\|\mathbf{m}_2 - \mathbf{m}_1\|_2 \cos\theta$
  2. Position: $\mathbf{p} = \mathbf{m}_2 + z_2 R(\pm\theta)\frac{\mathbf{m}_1 - \mathbf{m}_2}{\|\mathbf{m}_1 - \mathbf{m}_2\|_2}$
  3. Move with known $\mathbf{p}_\Delta, \theta_\Delta$ (in frame t)
  4. Orientation: $(\mathbf{p}_{t+1} - \mathbf{p}_t) = R(\theta_t)\mathbf{p}_\Delta$

- **3-D Localization**: given $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3 \in \mathbb{R}^3$ and $z_1, z_2, z_3 \in \mathbb{R}$
  1. Intersection of 2 circles with centers $\mathbf{o}_{12}, \mathbf{o}_{23}$, radii r_{12}, r_{23}, normals $\mathbf{n}_{12}, \mathbf{n}_{23}$ obtained via Law of Cosines and point $\mathbf{o}$ on intersecting line:

$$\begin{bmatrix} \mathbf{n}_{12}^\top \\ \mathbf{n}_{23}^\top \\ (\mathbf{n}_{12} \times \mathbf{n}_{23})^\top \end{bmatrix} \mathbf{o} = \begin{bmatrix} \mathbf{n}_{12}^\top \mathbf{o}_{12} \\ \mathbf{n}_{23}^\top \mathbf{o}_{23} \\ (\mathbf{n}_{12} \times \mathbf{n}_{23})^\top \mathbf{o}_{12} \end{bmatrix}$$

  2. Position: $\mathbf{p} = \mathbf{o}_{12} + r_{12} R(\mathbf{n}_{12}, \pm\theta)\frac{\mathbf{o} - \mathbf{o}_{12}}{\|\mathbf{o} - \mathbf{o}_{12}\|_2}$, where $\cos\theta = \frac{\|\mathbf{o} - \mathbf{o}_{12}\|_2}{r_{12}}$
  3. Move twice with known $\mathbf{p}_\Delta, R_\Delta, \mathbf{p}_{\Delta,2}, R_{\Delta,2}$
  4. Orientation: as long as $U := [\mathbf{p}_\Delta,\ R_\Delta \mathbf{p}_{\Delta,2},\ \mathbf{p}_\Delta \times R_\Delta \mathbf{p}_{\Delta,2})]$ is nonsingular:

$$R_t = [(\mathbf{p}_{t+1} - \mathbf{p}_t),\ (\mathbf{p}_{t+2} - \mathbf{p}_{t+1}),\ (\mathbf{p}_{t+1} - \mathbf{p}_t) \times (\mathbf{p}_{t+2} - \mathbf{p}_{t+1})]U^{-1}$$

- **Odometry**: not solvable