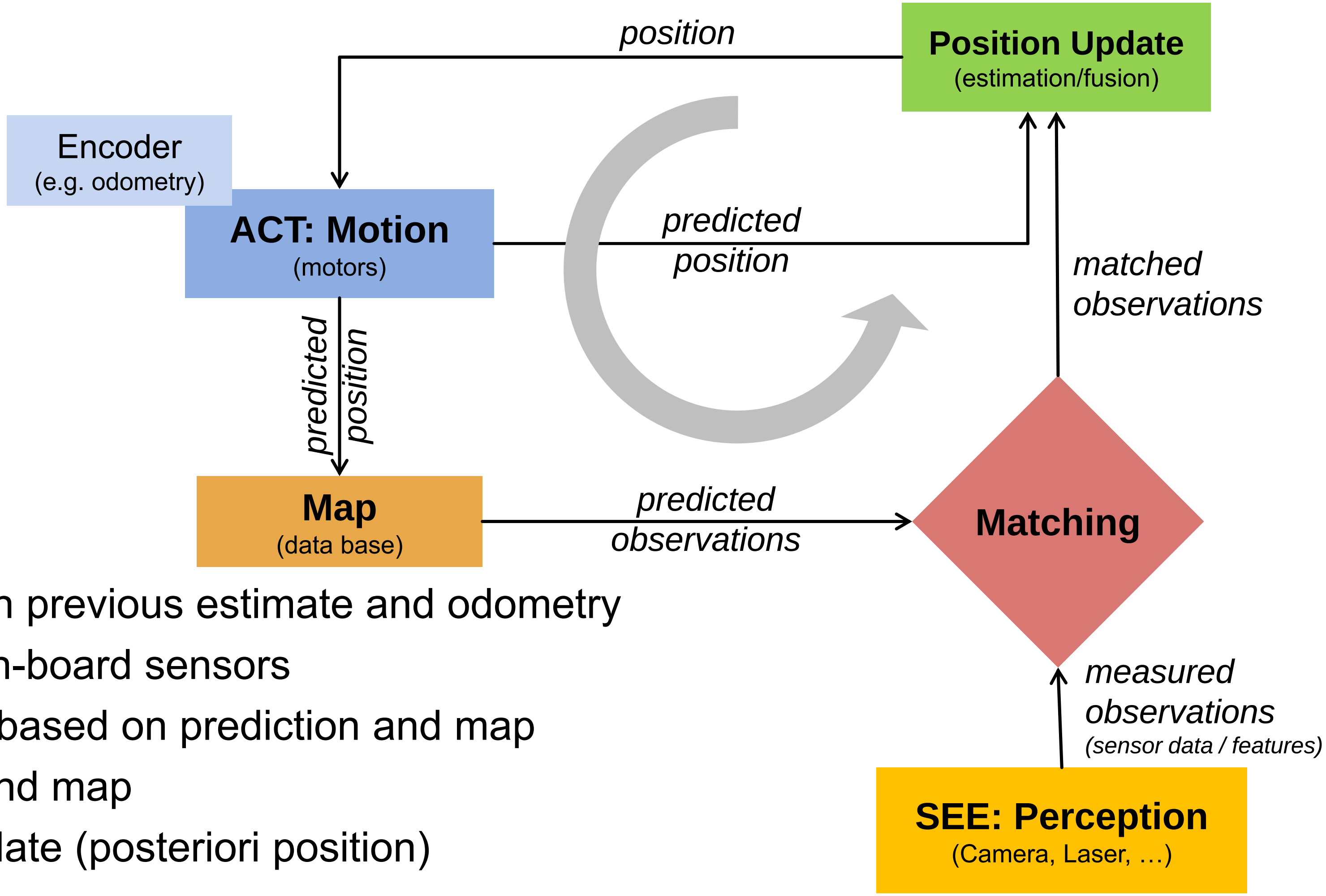
Localization | **the Kalman Filter Approach**
Autonomous Mobile Robots

Roland Siegwart
Margarita Chli, Paul Furgale, Marco Hutter, Martin Rufli, Davide Scaramuzza

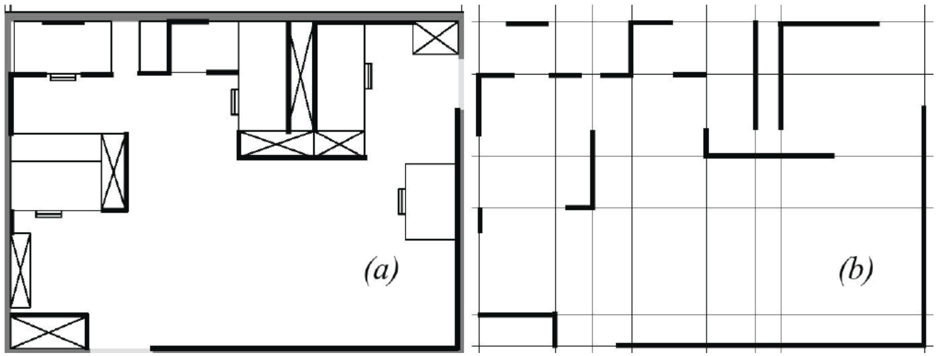
Kalman Filter Localization | applying probability theory to localization

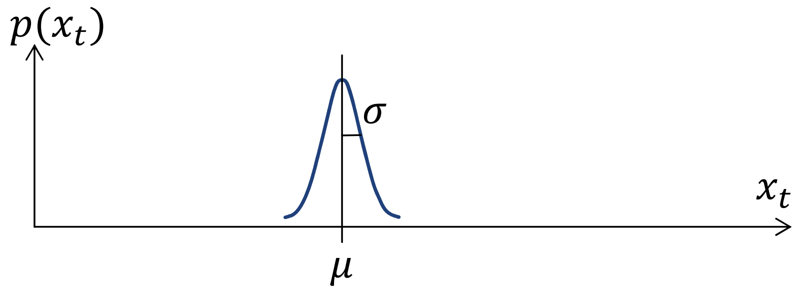


1. **Prediction (ACT)** based on previous estimate and odometry
2. **Observation (SEE)** with on-board sensors
3. **Measurement prediction** based on prediction and map
4. **Matching** of observation and map
5. **Estimation** → position update (posteriori position)

Kalman Filter Localization | Basics and assumption

- Continuous pose representation x_t
- Kalman Filter Assumptions:
  - Error approximation with normal distribution: $x = N(\mu, \sigma^2)$ (Gaussian model)
  - Output y_t distribution is a linear (or linearized) function of the input distribution: $y = Ax_1 + Bx_2$
- Kalman filter localization tracks the robot's belief state $p(x_t)$ typically as a single hypothesis with normal distribution.
- Kalman localization thus addresses the *position tracking problem*, but ***not*** the *global localization* or the *kidnapped robot problem*.





Kalman Filter Localization | prediction (odometry) - ACT

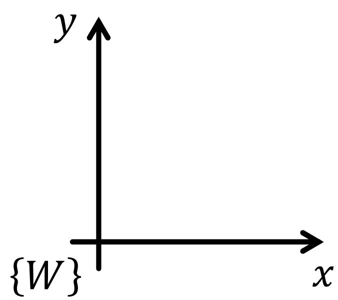
$$\hat{x}_t = f(x_{t-1}, u_t) = \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ \theta_{t-1} \end{bmatrix} + \begin{bmatrix} \frac{\Delta s_r + \Delta s_l}{2} \cos\left(\theta + \frac{\Delta s_r - \Delta s_l}{2b}\right) \\ \frac{\Delta s_r + \Delta s_l}{2} \sin\theta + \frac{\Delta s_r - \Delta s_l}{2b} \\ \frac{\Delta s_r - \Delta s_l}{b} \end{bmatrix}$$

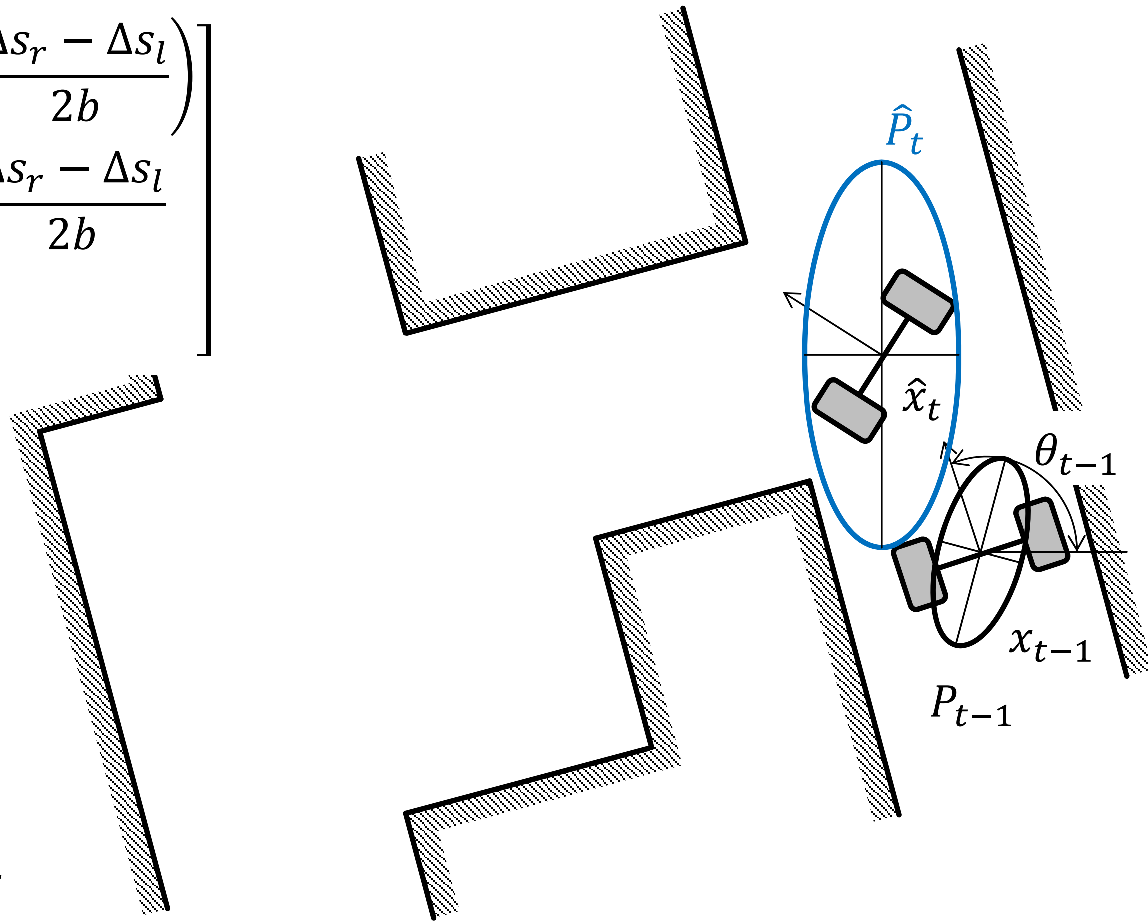
motion model

$$\hat{P}_t = F_x P_{t-1} {F_x}^T + F_u Q_t {F_u}^T$$

$$Q_t = \begin{bmatrix} k_r|\Delta s_r| & 0 \\ 0 & k_r|\Delta s_l| \end{bmatrix}$$

$F_x = \nabla_x f$ and $F_u = \nabla_x f$ represent the Jacobians of the function f with respect to x_t and u_t

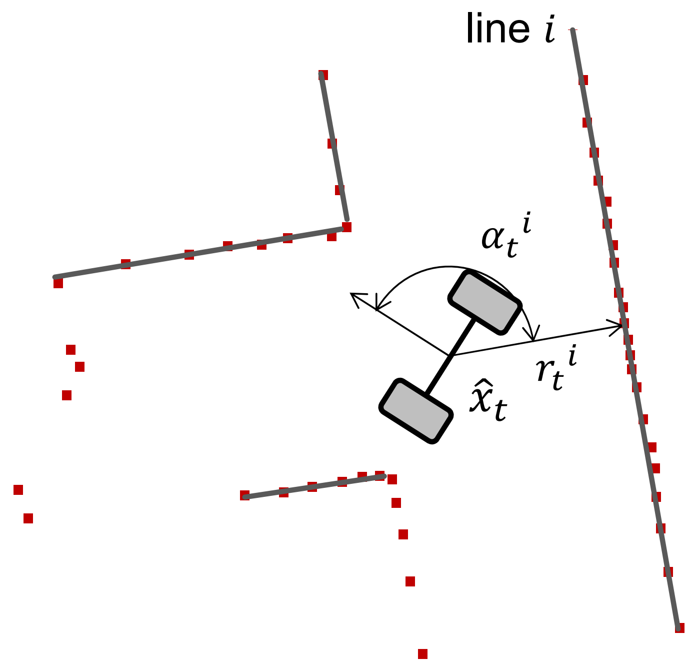


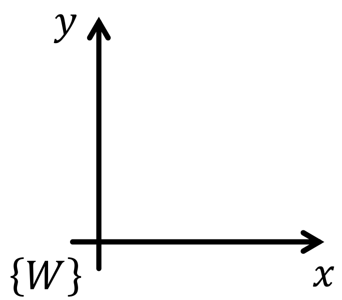


Kalman Filter Localization | observation - SEE

$$z_t{}^i = \begin{bmatrix} \alpha_t{}^i \\ r_t{}^i \end{bmatrix}$$

$$R_t{}^i = \begin{bmatrix} \sigma_{\alpha\alpha} & \sigma_{\alpha r} \\ \sigma_{r\alpha} & \sigma_{rr} \end{bmatrix}_t^i$$

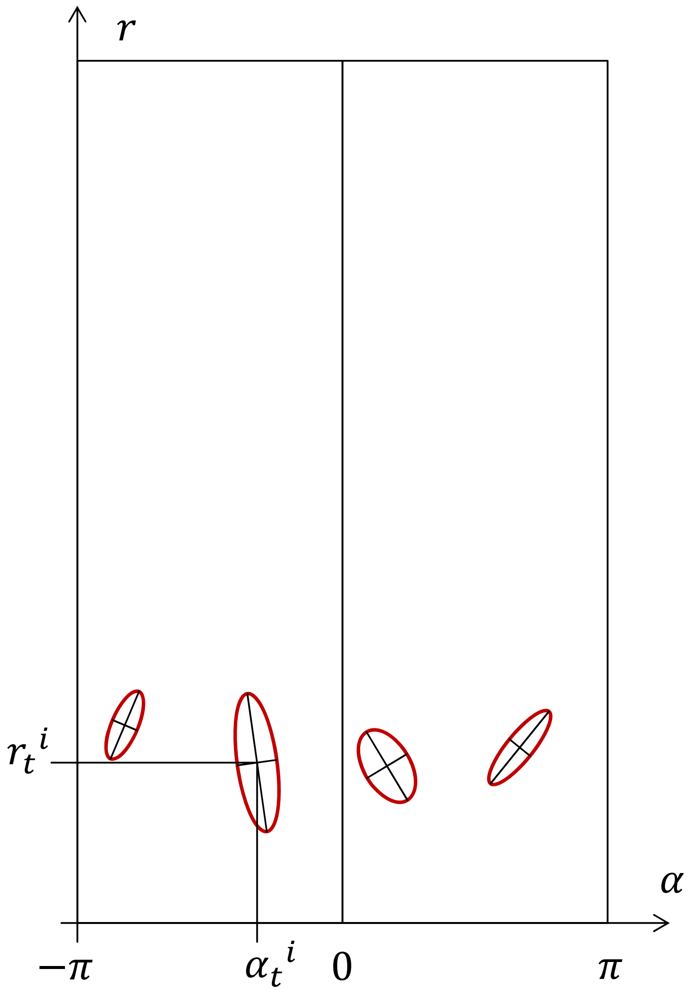


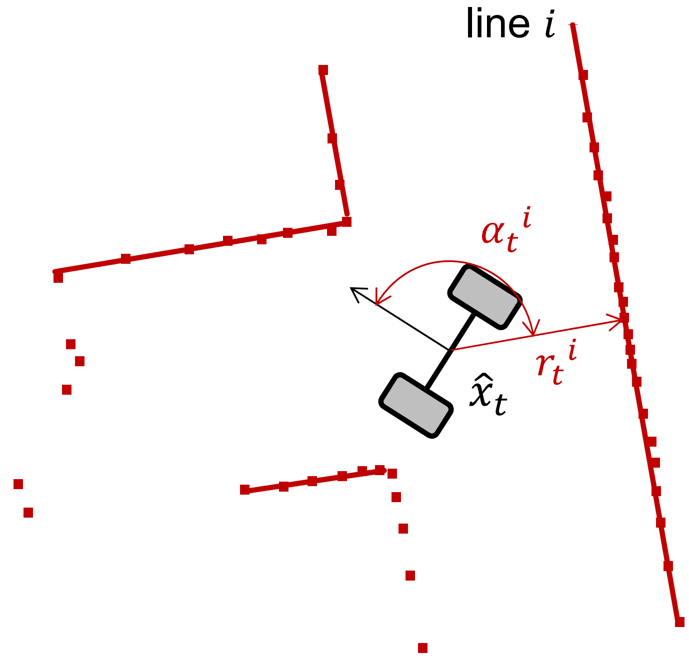


Kalman Filter Localization | observations in sensor model space

$$z_t{}^i = \begin{bmatrix} \alpha_t{}^i \\ r_t{}^i \end{bmatrix}$$

$$R_t{}^i = \begin{bmatrix} \sigma_{\alpha\alpha} & \sigma_{\alpha r} \\ \sigma_{r\alpha} & \sigma_{rr} \end{bmatrix}_t^i$$

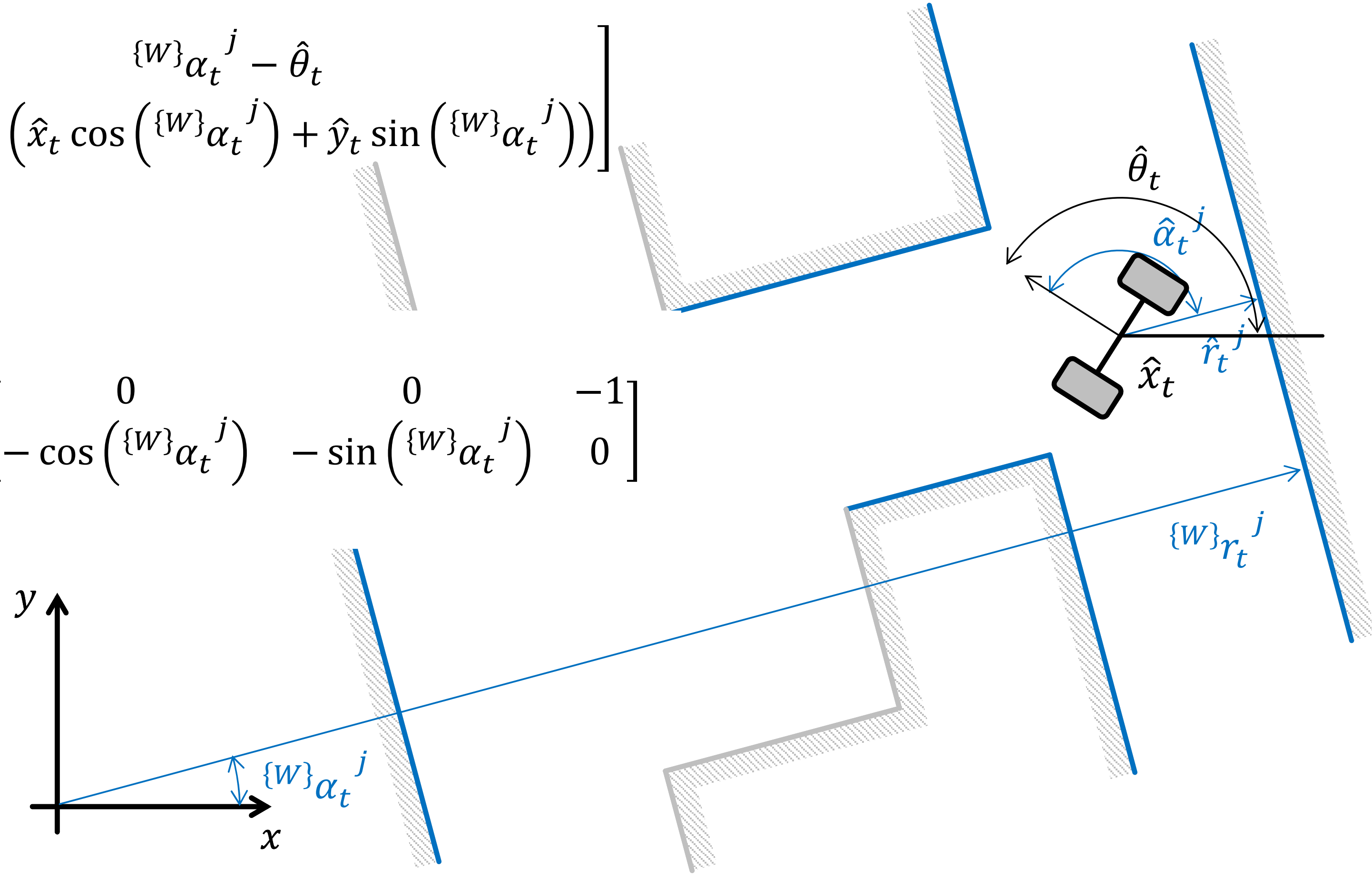




Kalman Filter Localization | measurement prediction

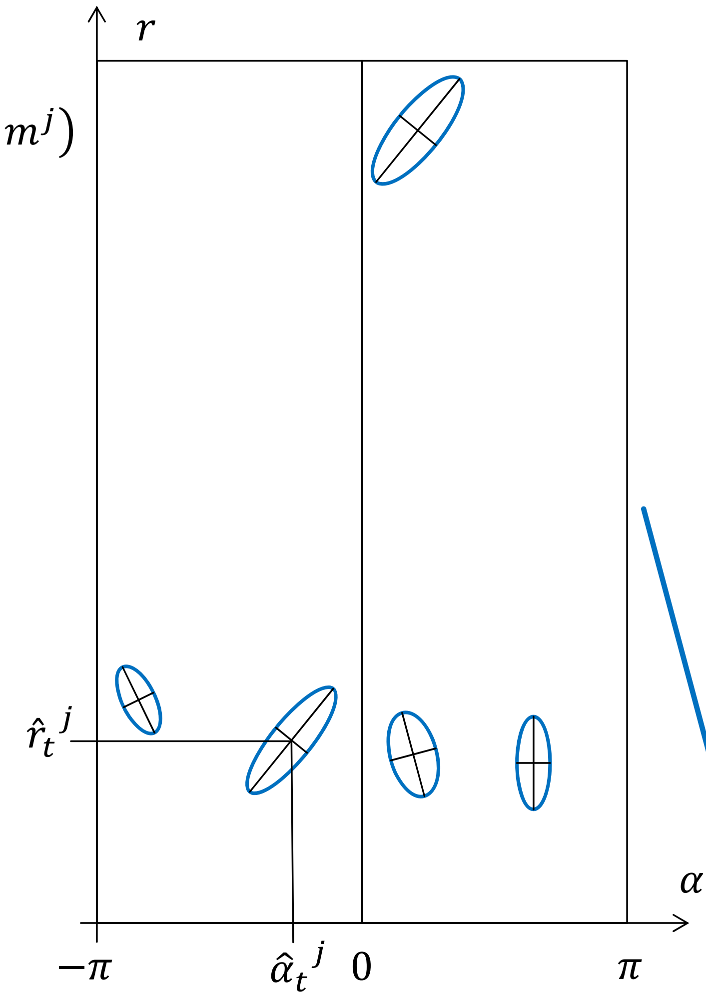
$$\hat{z}_t{}^j = \begin{bmatrix} \hat{\alpha}_t{}^j \\ \hat{r}_t{}^j \end{bmatrix} = h^j(\hat{x}_t, m^j) = \begin{bmatrix} {}^{\{W\}}\alpha_t{}^j - \hat{\theta}_t \\ {}^{\{W\}}r_t{}^j - \left(\hat{x}_t \cos\left({}^{\{W\}}\alpha_t{}^j\right) + \hat{y}_t \sin\left({}^{\{W\}}\alpha_t{}^j\right)\right) \end{bmatrix}$$

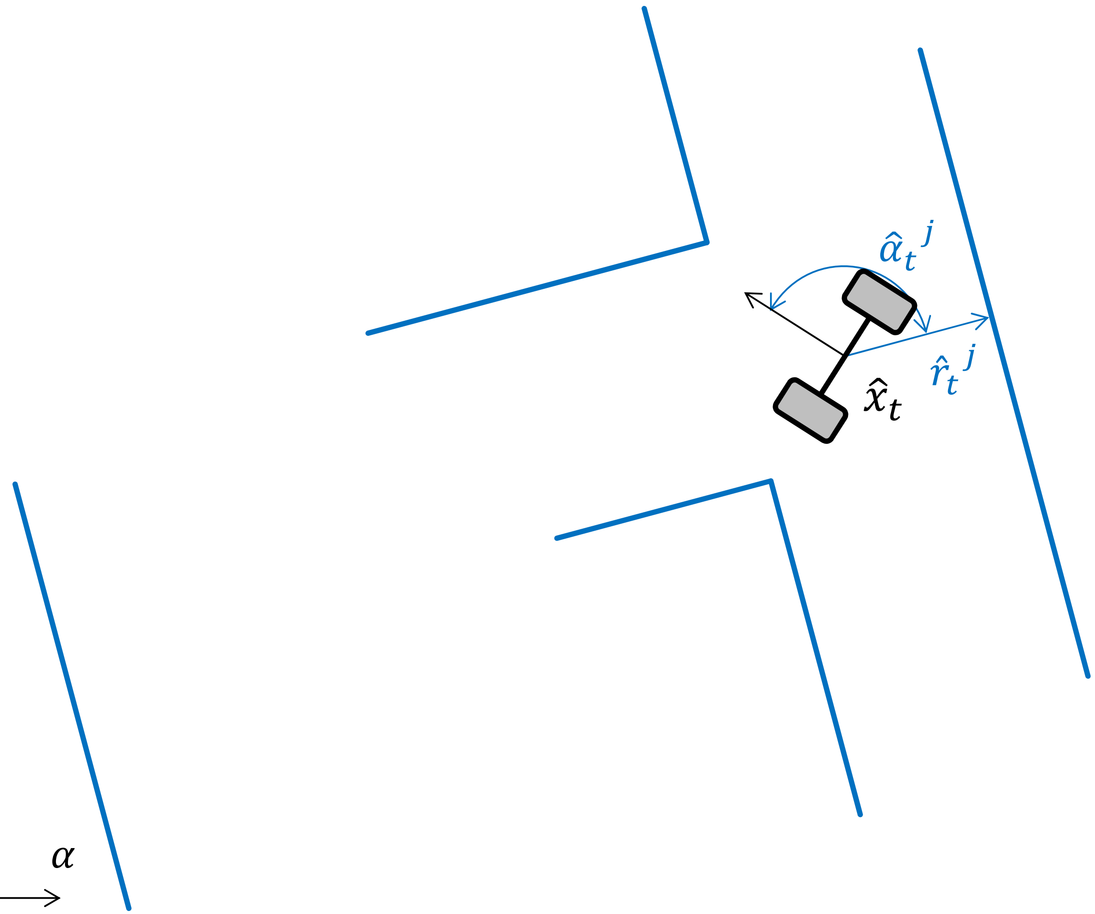
$$H^j = \nabla h^j = \begin{bmatrix} \frac{\partial \alpha_t{}^i}{\partial \hat{x}} & \frac{\partial \alpha_t{}^i}{\partial \hat{y}} & \frac{\partial \alpha_t{}^i}{\partial \hat{\theta}} \\ \frac{\partial r_t{}^i}{\partial \hat{x}} & \frac{\partial r_t{}^i}{\partial \hat{y}} & \frac{\partial r_t{}^i}{\partial \hat{\theta}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 \\ -\cos\left({}^{\{W\}}\alpha_t{}^j\right) & -\sin\left({}^{\{W\}}\alpha_t{}^j\right) & 0 \end{bmatrix}$$



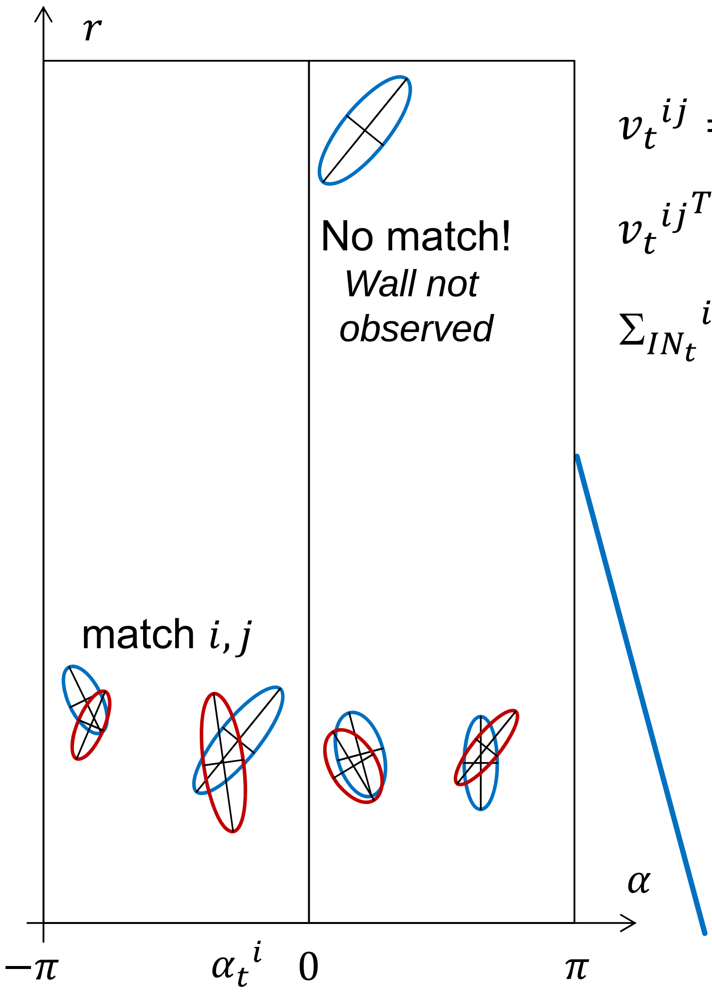
Kalman Filter Localization | measurement prediction in model space

$$\hat{z}_t^{\,j} = \begin{bmatrix} \hat{\alpha}_t^{\,j} \\ \hat{r}_t^{\,j} \end{bmatrix} = h^j(\hat{x}_t, m^j)$$





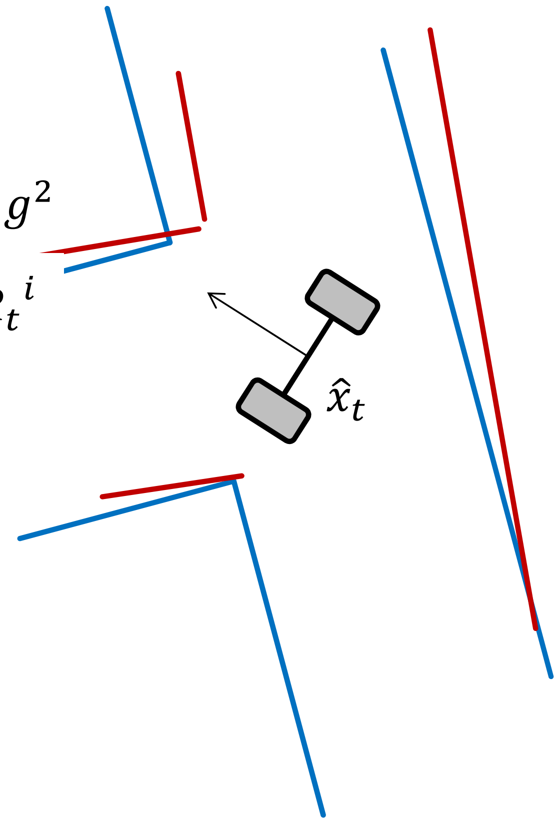
Kalman Filter Localization | matching in sensor model space



$$v_t{}^{ij} = \left[z_t{}^i - \hat{z}_t{}^j\right]$$

$$v_t{}^{ij\,T} \cdot \left(\Sigma_{IN_t}{}^{ij}\right)^{-1} \cdot v_t{}^{ij} \leq g^2$$

$$\Sigma_{IN_t}{}^{ij} = H^j \cdot \hat{P}_t \cdot H^{j\,T} + R_t{}^i$$



Kalman Filter Localization | estimation

- For each found match we can now estimate an position update:

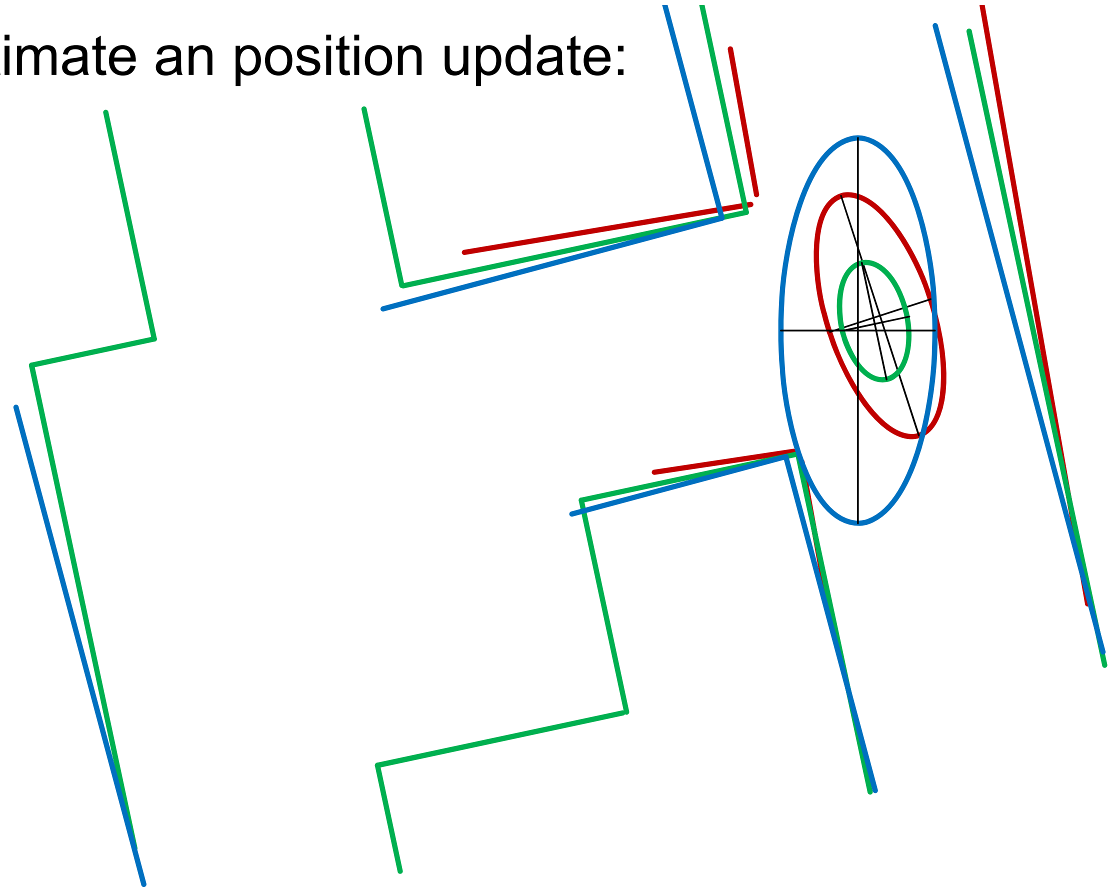
$$x_t = \hat{x}_t + K_t v_t$$

- where $K_t = \hat{P}_t {H_t}^T \left(\Sigma_{IN_t}\right)^{-1}$

  is the Kalman gain

- and the corresponding position covariance P_t :

$$P_t = \hat{P}_t - K_t \Sigma_{IN_t} {K_t}^T$$



Kalman Filter Localization | in summery

1. **Prediction (ACT)** based on previous estimate and odometry
2. **Observation (SEE)** with on-board sensors
3. **Measurement prediction** based on prediction and map
4. **Matching** of observation and map
5. **Estimation** → position update (posteriori position)