Introduction to Mobile Robotics

Bayes Filter – Extended Kalman Filter

Daniel Büscher

Bayes Filter Reminder

$$bel(x_t) = \eta \, p(z_t \mid x_t) \int p(x_t \mid u_t, x_{t-1}) \, bel(x_{t-1}) \, dx_{t-1}$$

- Prediction

$$\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1}) \, bel(x_{t-1}) \, dx_{t-1}$$

- Correction

$$bel(x_t) = \eta \, p(z_t \mid x_t) \, \overline{bel}(x_t)$$

Discrete Kalman Filter

Estimates the state x of a discrete-time controlled process

$$x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t$$

with a measurement

$$z_t = C_t x_t + \delta_t$$

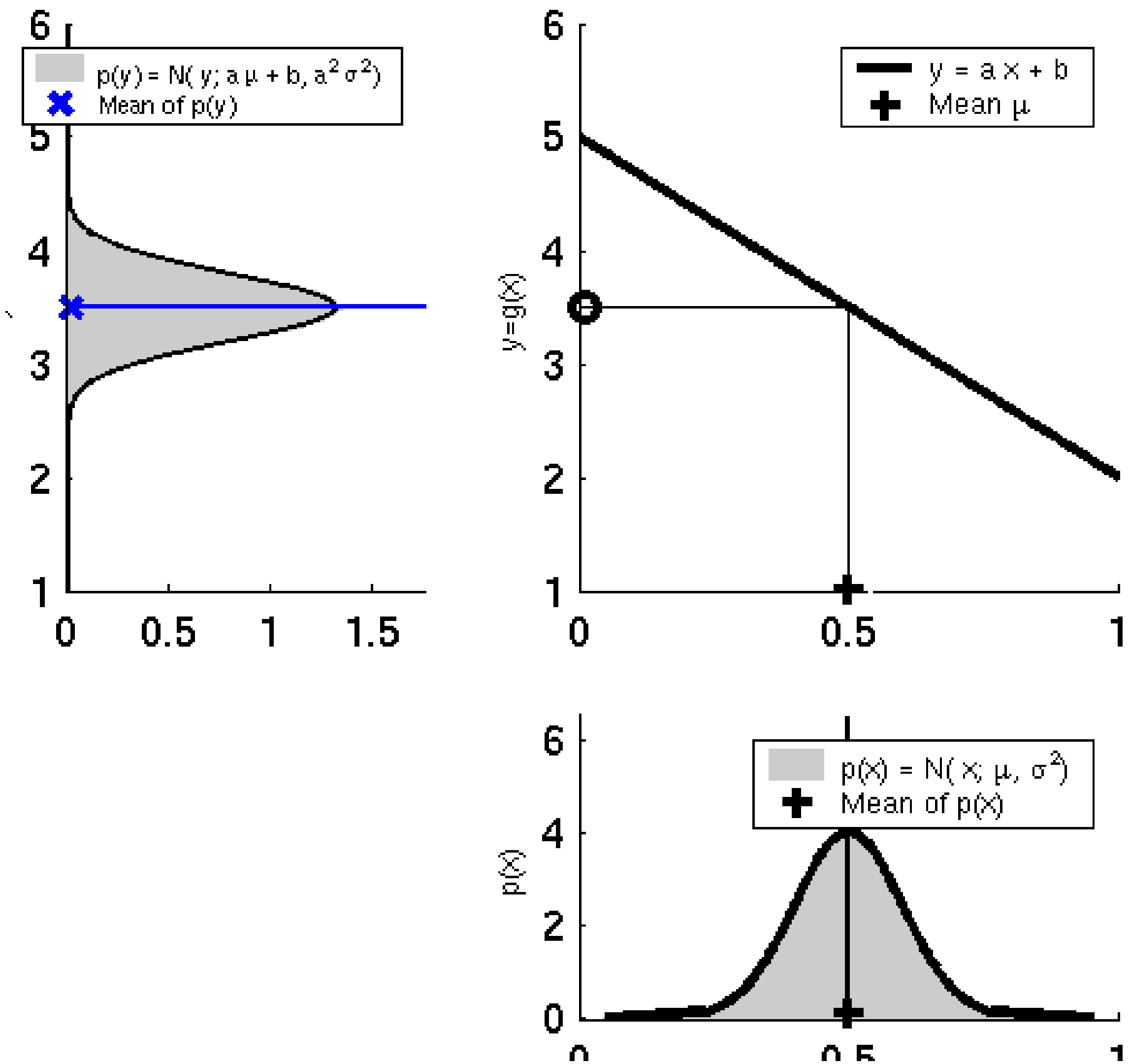
Nonlinear Dynamic Systems

- Most realistic robotic problems involve nonlinear functions

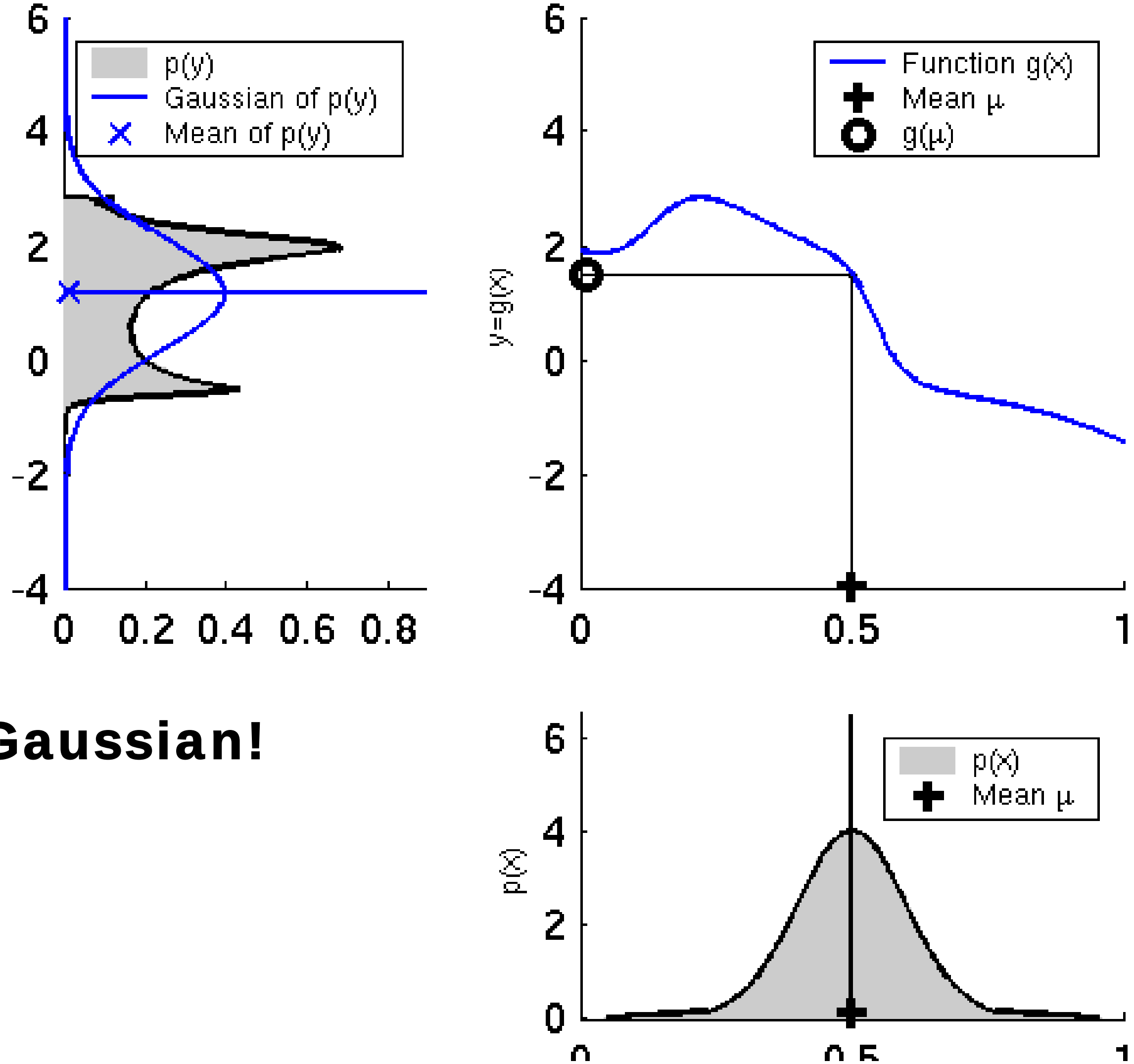
~~$x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t$~~ ➡ $x_t = g(u_t, x_{t-1})$

~~$z_t = C_t x_t + \delta_t$~~ ➡ $z_t = h(x_t)$

Linear Fn: Gaussian preserved



Non-Linear Function



Non-Gaussian!

Non-Gaussian Distributions

- Non-linear functions lead to non-Gaussian distributions
- The Kalman filter is not applicable anymore!

What can be done to resolve this?

Local linearization!

1. Linearization

EKF Linearization: First Order Taylor Expansion

- Prediction:

$$g(u_t, x_{t-1}) \approx g(u_t, \mu_{t-1}) + \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} (x_{t-1} - \mu_{t-1})$$

$$= g(u_t, \mu_{t-1}) + G_t \, (x_{t-1} - \mu_{t-1})$$

- Correction:

$$h(x_t) \approx h(\bar{\mu}_t) + \frac{\partial h(\bar{\mu}_t)}{\partial x_t} (x_t - \bar{\mu}_t)$$

$$= h(\bar{\mu}_t) + H_t \, (x_t - \bar{\mu}_t)$$

Jacobian matrices: linear functions!

Jacobian Matrix

- Non-square $n \times m$ in general
- Given a vector-valued function

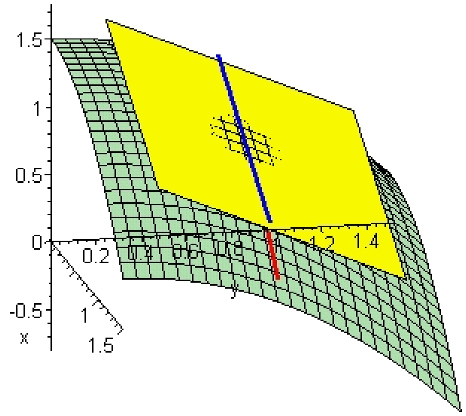
$$f(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{bmatrix}$$

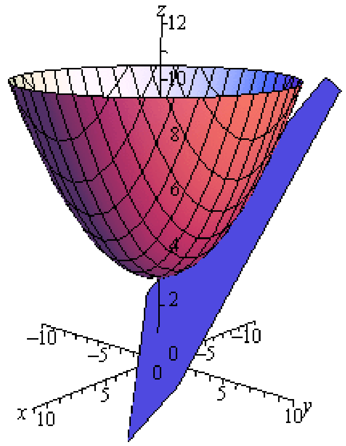
it is defined as

$$\mathbf{F_x} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

Jacobian Matrix interpretation

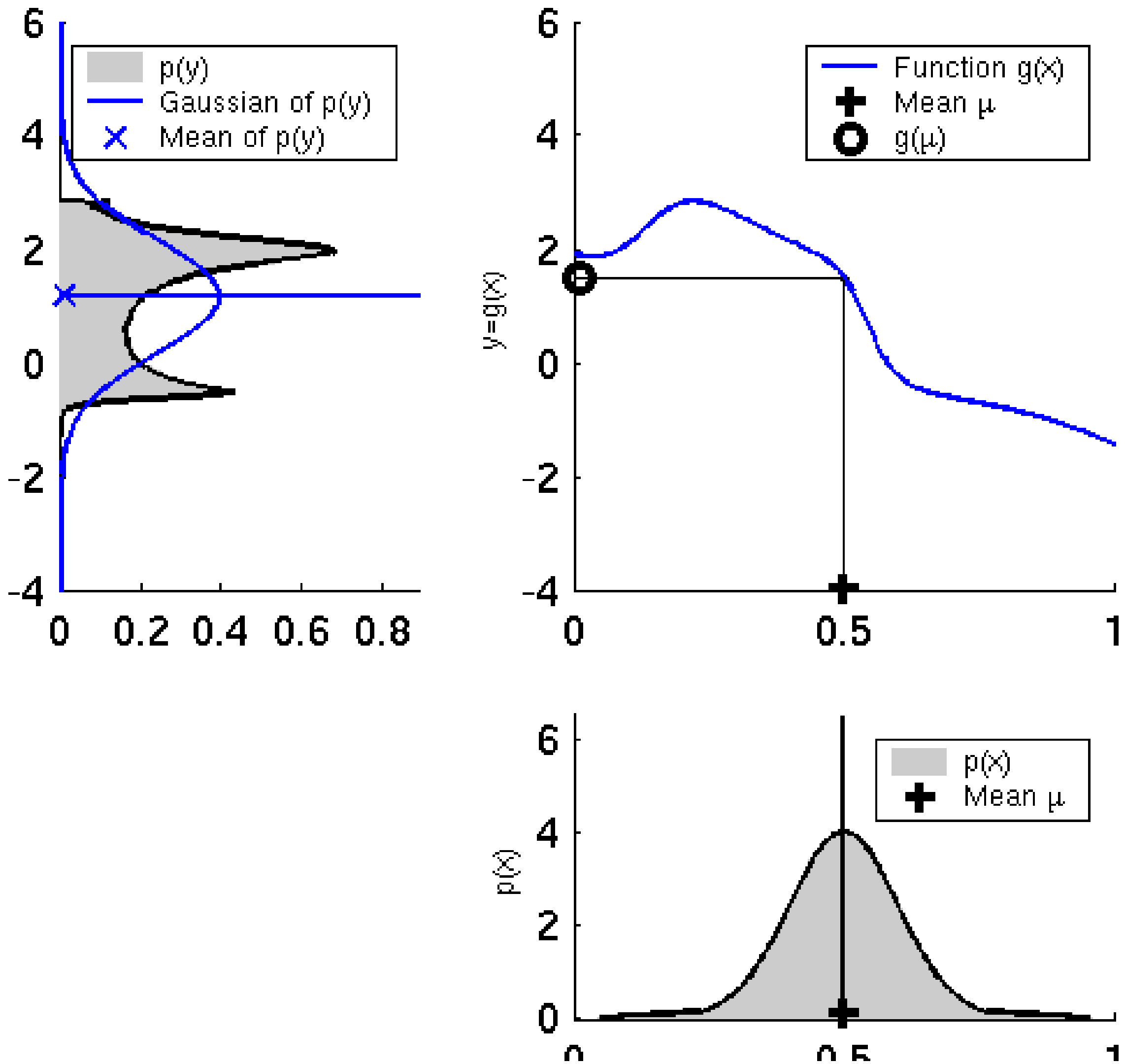
- It is the orientation of the tangent plane to the vector-valued function at a given point



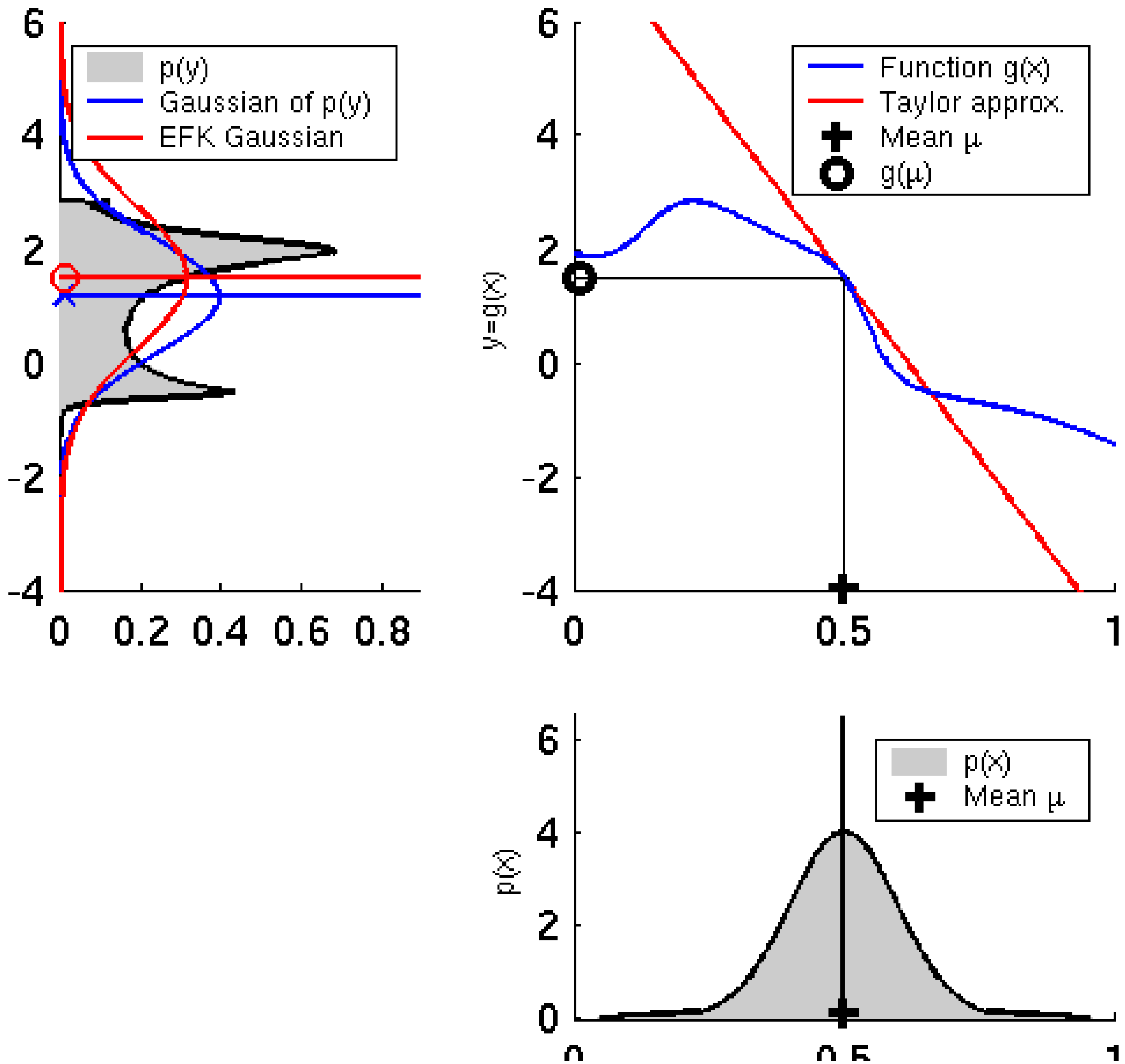


- Generalizes the gradient of a scalar-valued function

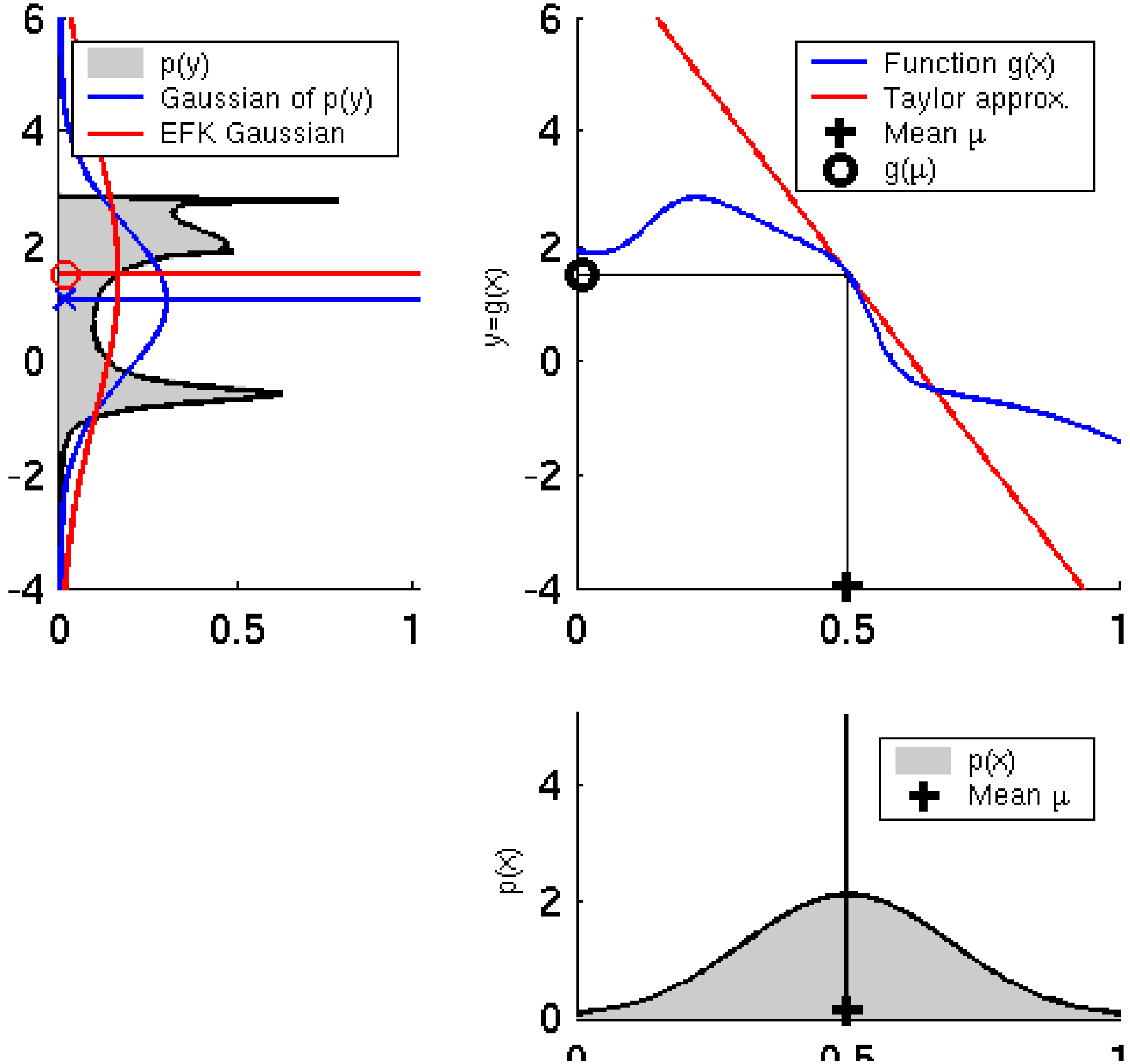
Non-Linear Function



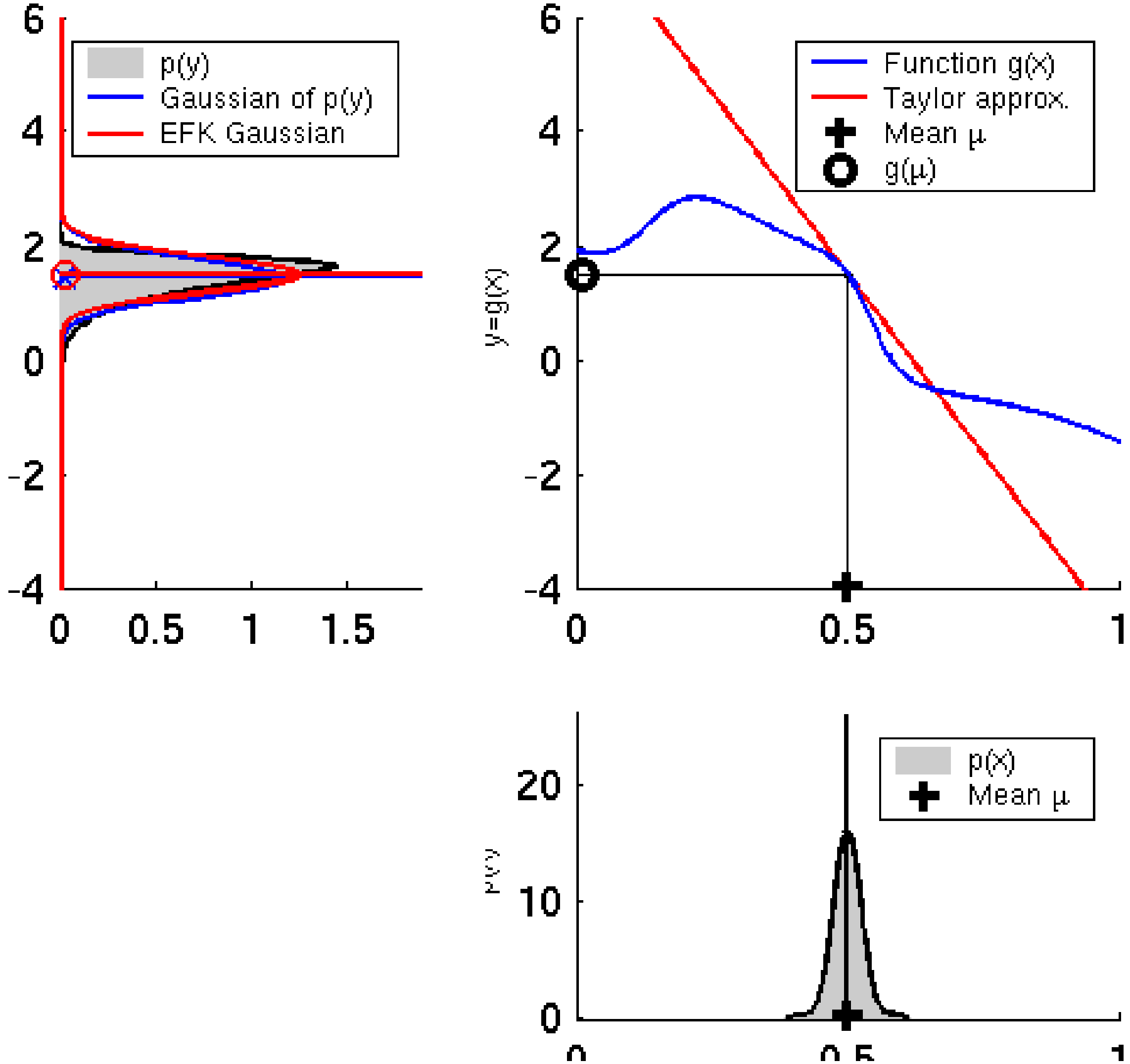
EKF Linearization (1)



EKF Linearization (2)



EKF Linearization (3)



2. EKF Algorithm

Reminder: KF Algorithm

1. Algorithm **Kalman_filter**(μ_{t-1}, Σ_{t-1}, u_t, z_t):

   Prediction:

2. $\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$
3. $\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + Q_t$

   Correction:

4. $K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + R_t)^{-1}$
5. $\mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t)$
6. $\Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$

7. Return μ_t, Σ_t

EKF Algorithm

1. **Extended_Kalman_filter**(μ_{t-1}, Σ_{t-1}, u_t, z_t):

   Prediction:

2. $\bar{\mu}_t = g(u_t, \mu_{t-1})$
3. $\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + Q_t$

   Correction:

4. $K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + R_t)^{-1}$
5. $\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t))$
6. $\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$
7. Return μ_t, Σ_t

Kalman_filter:

$$\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$$

$$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + Q_t$$

$$K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + R_t)^{-1}$$

$$\mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t)$$

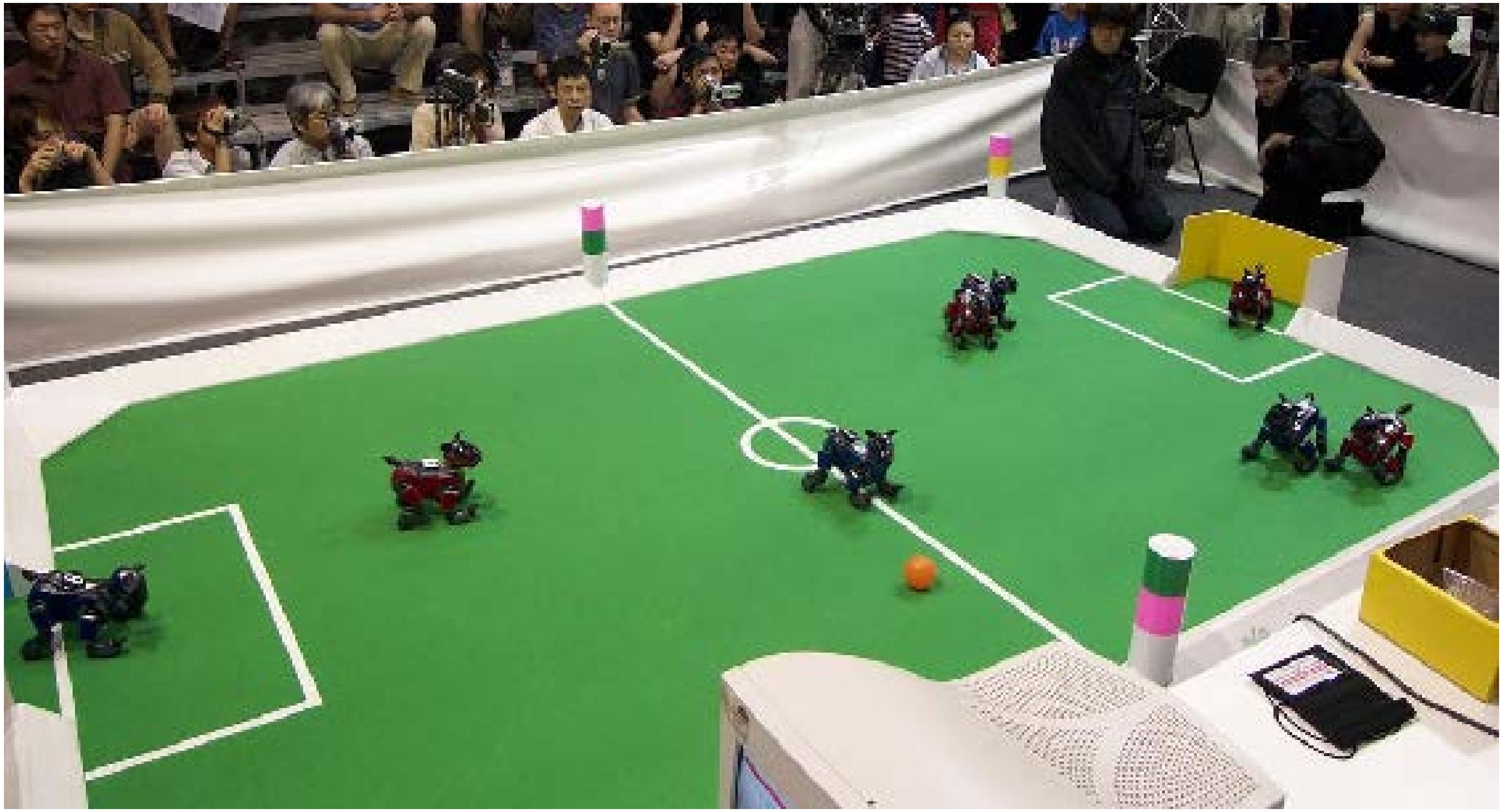
$$\Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$$

$$G_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} \qquad H_t = \frac{\partial h(\bar{\mu}_t)}{\partial x_t}$$

3. EKF Example

Example: EKF Localization

- EKF localization with landmarks (point features)



1. **EKF_localization** (μ_{t-1}, Σ_{t-1}, u_t, z_t, m):

Prediction:

2. $$G_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial \mu_{t-1}} = \begin{pmatrix} \frac{\partial x'}{\partial \mu_{t-1,x}} & \frac{\partial x'}{\partial \mu_{t-1,y}} & \frac{\partial x'}{\partial \mu_{t-1,\theta}} \\ \frac{\partial y'}{\partial \mu_{t-1,x}} & \frac{\partial y'}{\partial \mu_{t-1,y}} & \frac{\partial y'}{\partial \mu_{t-1,\theta}} \\ \frac{\partial \theta'}{\partial \mu_{t-1,x}} & \frac{\partial \theta'}{\partial \mu_{t-1,y}} & \frac{\partial \theta'}{\partial \mu_{t-1,\theta}} \end{pmatrix}$$ Jacobian of g w.r.t location

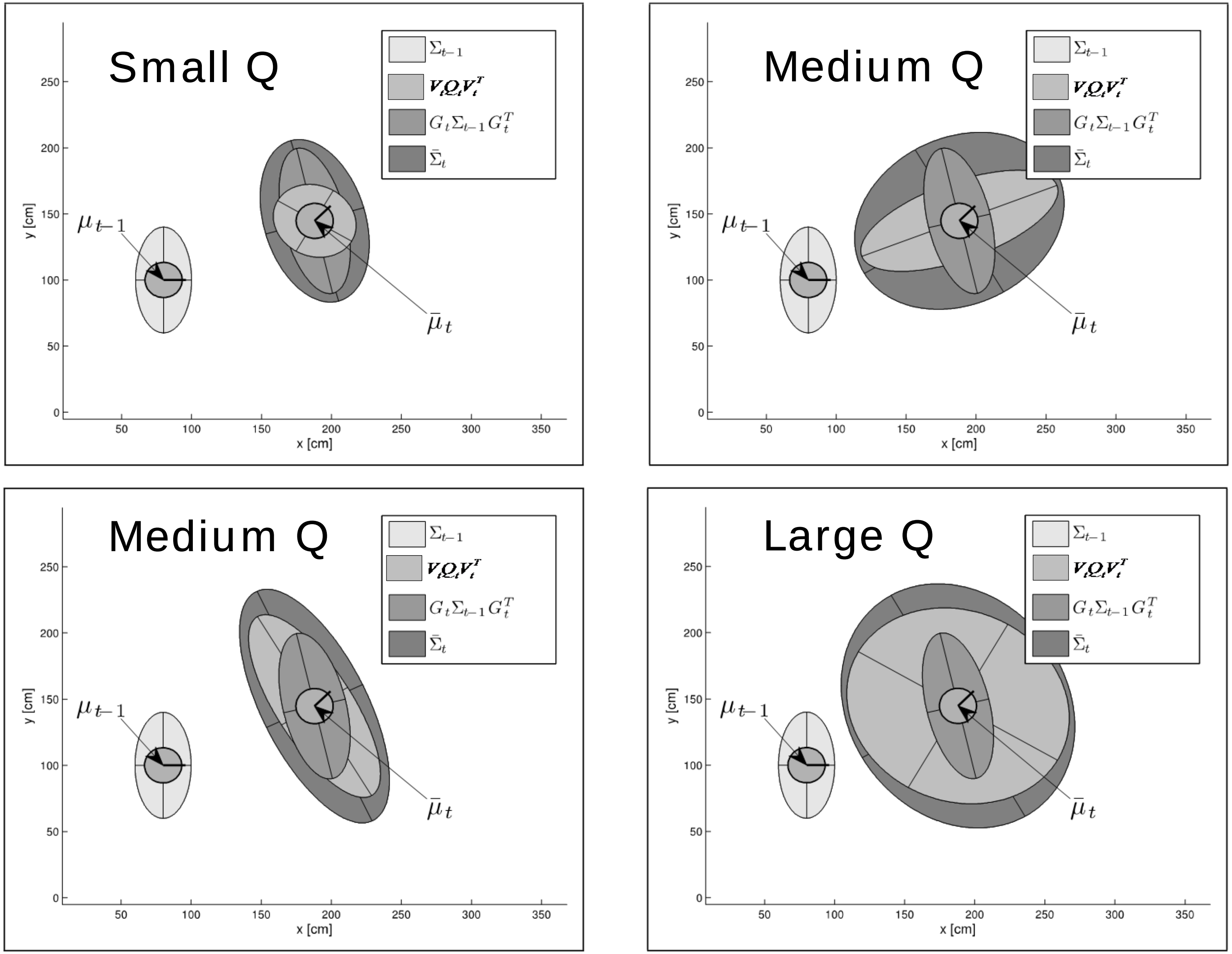
3. $$V_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial u_t} = \begin{pmatrix} \frac{\partial x'}{\partial v_t} & \frac{\partial x'}{\partial \omega_t} \\ \frac{\partial y'}{\partial v_t} & \frac{\partial y'}{\partial \omega_t} \\ \frac{\partial \theta'}{\partial v_t} & \frac{\partial \theta'}{\partial \omega_t} \end{pmatrix}$$ Jacobian of g w.r.t control

4. $$Q_t = \begin{pmatrix} (\alpha_1 |v_t| + \alpha_2 |\omega_t|)^2 & 0 \\ 0 & (\alpha_3 |v_t| + \alpha_4 |\omega_t|)^2 \end{pmatrix}$$ Motion noise

5. $$\bar{\mu}_t = g(u_t, \mu_{t-1})$$ Predicted mean

6. $$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + V_t Q_t V_t^T$$ Predicted covariance (V maps Q into state space)

EKF Prediction Step



Correction: (EKF_localization continued)

7. $\hat{z}_t = \begin{pmatrix} \sqrt{(m_x - \bar{\mu}_{t,x})^2 + (m_y - \bar{\mu}_{t,y})^2} \\ \operatorname{atan2}(m_y - \bar{\mu}_{t,y}, m_x - \bar{\mu}_{t,x}) - \bar{\mu}_{t,\theta} \end{pmatrix}$ Predicted measurement mean $h(\mu) = z = (r, \Phi)^T$

8. $H_t = \frac{\partial h(\bar{\mu}_t, m)}{\partial x_t} = \begin{pmatrix} \frac{\partial r_t}{\partial \bar{\mu}_{t,x}} & \frac{\partial r_t}{\partial \bar{\mu}_{t,y}} & \frac{\partial r_t}{\partial \bar{\mu}_{t,\theta}} \\ \frac{\partial \phi_t}{\partial \bar{\mu}_{t,x}} & \frac{\partial \phi_t}{\partial \bar{\mu}_{t,y}} & \frac{\partial \phi_t}{\partial \bar{\mu}_{t,\theta}} \end{pmatrix}$ Jacobian of h w.r.t location

9. $R_t = \begin{pmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\Phi^2 \end{pmatrix}$ Measurement noise

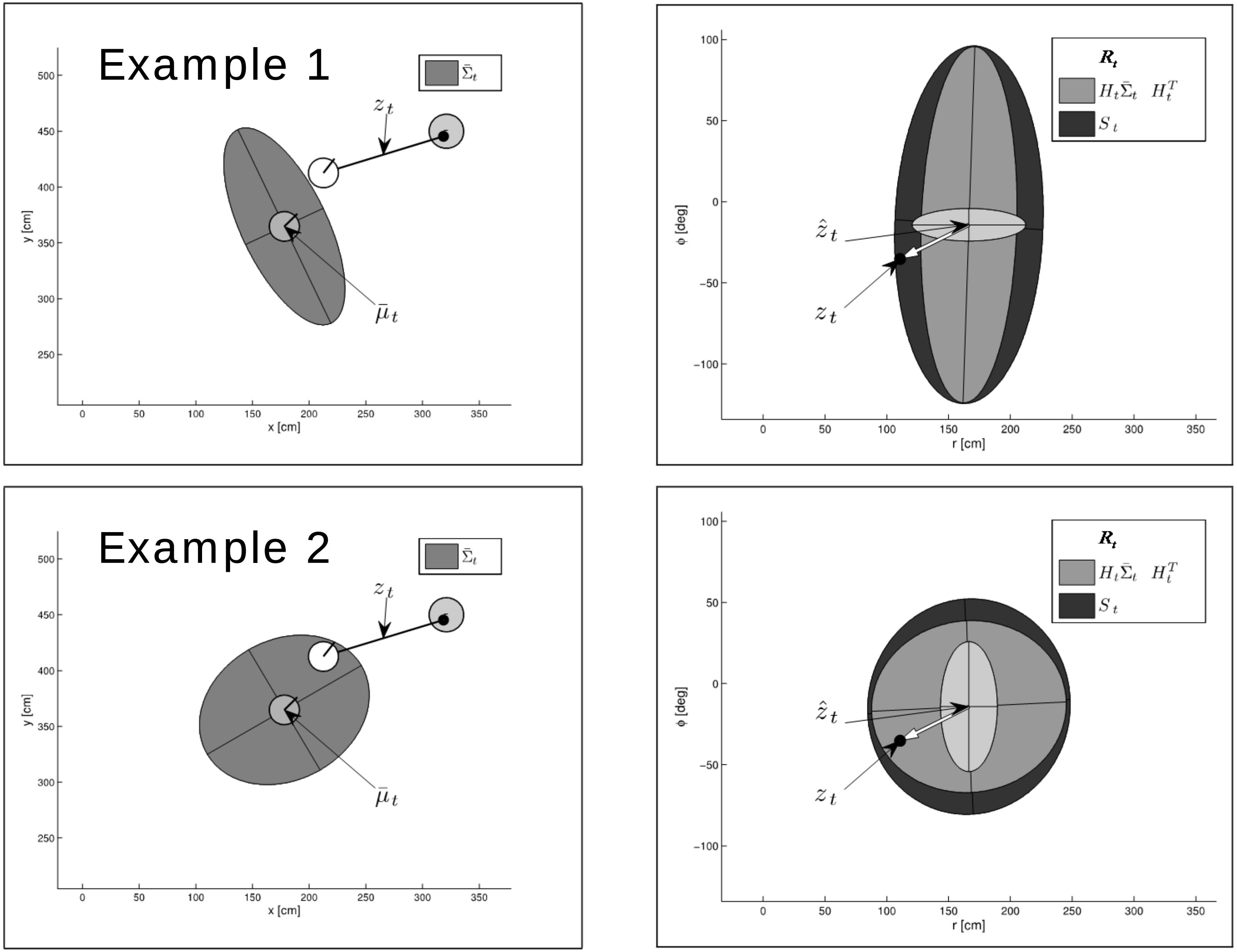
10. $S_t = H_t \bar{\Sigma}_t H_t^T + R_t$ Innovation covariance

11. $K_t = \bar{\Sigma}_t H_t^T S_t^{-1}$ Kalman gain

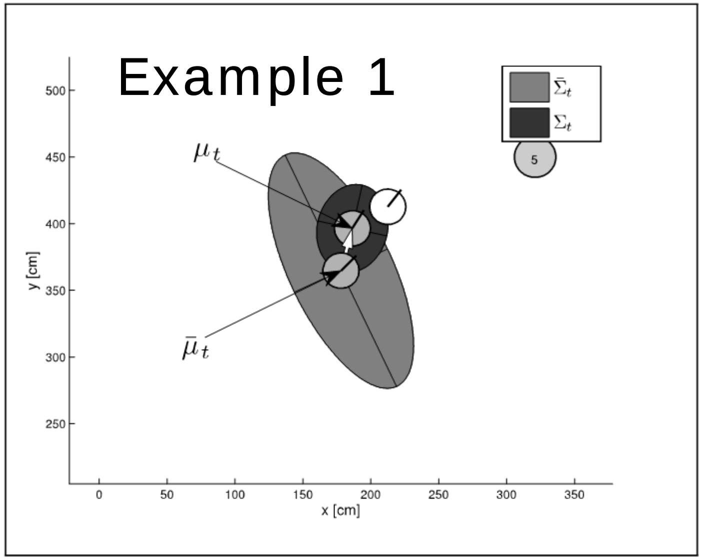
12. $\mu_t = \bar{\mu}_t + K_t (z_t - \hat{z}_t)$ Updated mean

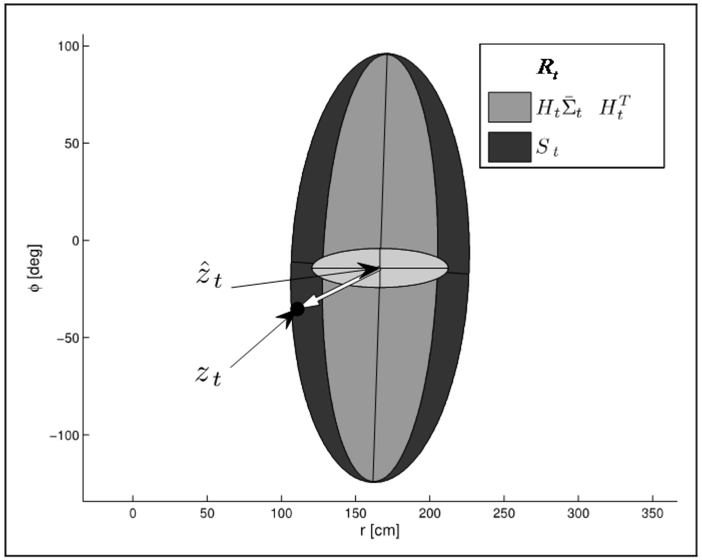
13. $\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$ Updated covariance

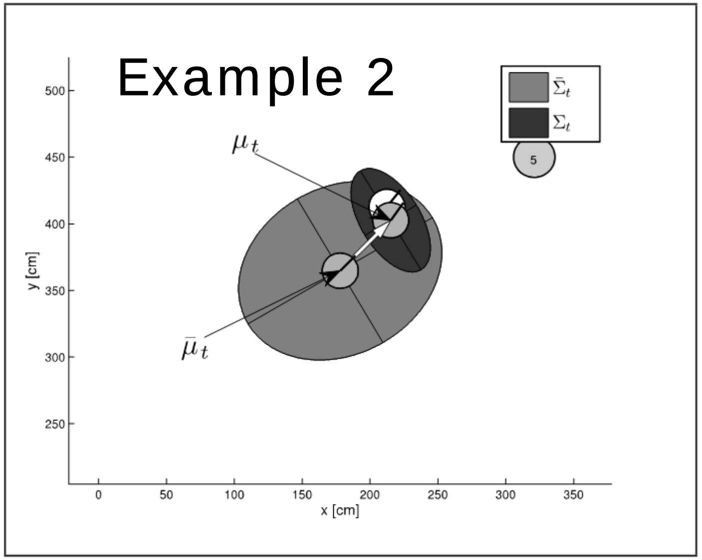
EKF Observation Prediction

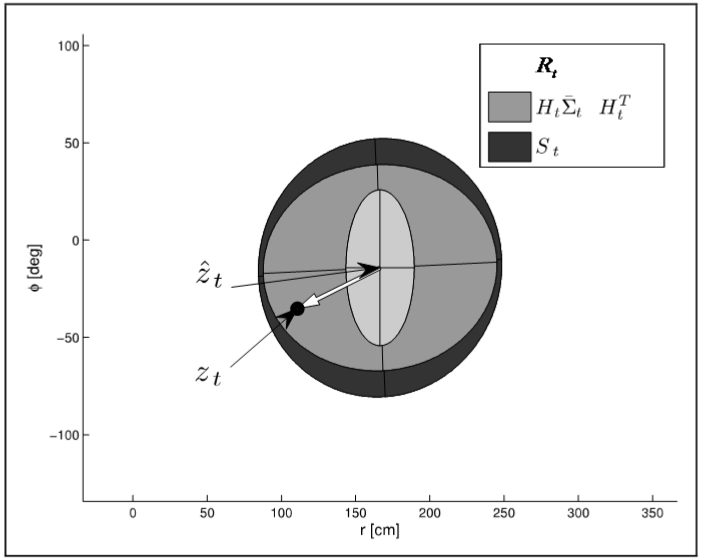


EKF Correction Step

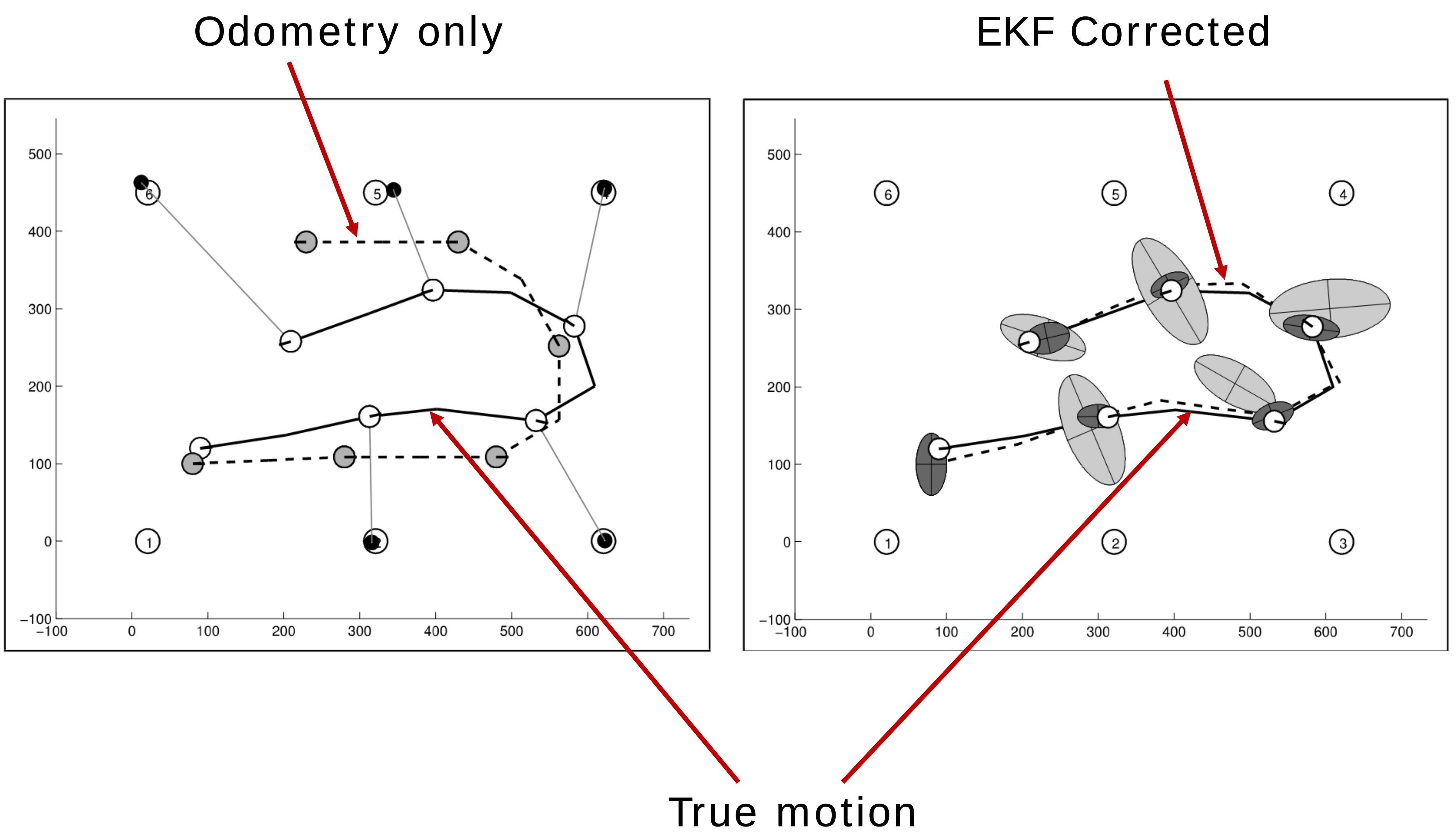








Localization Sequence



EKF Summary

- The Extended Kalman Filter is an ad-hoc solution to deal with non-linearities
- Performs local linearization in each step
- Works well in practice for moderate non-linearities (example: landmark localization)
- It is optimal if the measurement and the motion model are both linear (reduces to the KF)
- There exist better ways for dealing with non-linearities, such as the unscented Kalman filter