Using the Kalman filter
Extended Kalman filter

Doz. G. Bleser
Prof. Stricker

Computer Vision: Object and People Tracking
Outline

• Recap: Kalman filter algorithm
• Using Kalman filters
• Extended Kalman filter algorithm
Recap: Kalman filter

- Bayes update rule + linear Gaussian models $\Rightarrow$ Kalman filter

\[
bel(x_t) = \eta p(z_t | x_t) \int p(x_t | x_{t-1}, u_t) bel(x_{t-1}) dx_{t-1}
\]

- Posterior
- Likelihood
- Motion model
- Posterior at $t-1$
Recap: Kalman filter algorithm

1. **Kalman_filter**\( (\mu_{t-1}, \Sigma_{t-1}, u_t, z_t)\):

2. Prediction:
   \[
   \mu_t = A_t \mu_{t-1} + B_t u_t
   \]
   \[
   \Sigma_t = A_t \Sigma_{t-1} A_t^T + R_t
   \]

3. Requirements:
   - Initial belief is Gaussian
   - Linear Gaussian state-space model

5. Correction:
6. \[
   K_t = \Sigma_t C_t^T (C_t \Sigma_t C_t^T + Q_t)^{-1}
   \]
7. \[
   \mu_t = \mu_t + K_t (z_t - C_t \mu_t)
   \]
8. \[
   \Sigma_t = (I - K_t C_t) \Sigma_t
   \]
9. Return \( \mu_t, \Sigma_t \)
Using Kalman filters

Application to a simple 2D tracking problem

Some slides based on K. Smith
Track an aircraft in a video sequence

- Assumptions:
  - Image processing provides noisy 2D positions of aircraft
  - Motion model is const. pos.
  → Linear Gaussian system

- Formulate the state-space model:

\[
x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t
\]
\[
\varepsilon_t \sim N(0, R_t)
\]
\[
z_t = C_t x_t + \delta_t
\]
\[
\delta_t \sim N(0, Q_t)
\]
Track an aircraft in a video sequence

- State: 2D image position
  \[ x_t = \begin{pmatrix} x \\ y \end{pmatrix} \]

- Measurement: 2D image position
  \[ z_t = \begin{pmatrix} x \\ y \end{pmatrix} \]
Track an aircraft in a video sequence

- **Motion/dynamic model:**
  \[ x_t = Ax_{t-1} + Bu_t + \varepsilon_t \]
  \[
  A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad Bu_t = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\
  R = \begin{pmatrix} \sigma_{\varepsilon}^2 & 0 \\ 0 & \sigma_{\varepsilon}^2 \end{pmatrix}, \varepsilon_t \sim N(0, R)
  \]

- **Measurement model:**
  \[ z_t = C_t x_t + \delta_t \]
  \[
  C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\
  Q = \begin{pmatrix} \sigma_{\delta}^2 & 0 \\ 0 & \sigma_{\delta}^2 \end{pmatrix}, \delta_t \sim N(0, Q)\]
1. **Kalman_filter**($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

2. Prediction:

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

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

5. Correction:

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

7. $\mu_t = \mu_t + K_t (z_t - C_t \mu_t)$

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

9. Return $\mu_t, \Sigma_t$

Note: $\bar{x}_t := x_{t|t-1}$

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad Bu_t = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$R = \begin{pmatrix} \sigma^2_{\varepsilon} & 0 \\ 0 & \sigma^2_{\varepsilon} \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad Q = \begin{pmatrix} \sigma^2_{\delta} & 0 \\ 0 & \sigma^2_{\delta} \end{pmatrix}$$
Track an aircraft in a video sequence

• Prediction: \( \mu_{t|t-1} = \mu_{t-1} \quad \Sigma_{t|t-1} = \Sigma_{t-1} + R \)
• Predicted measurement = predicted state
• Measurement update:
  \[
  K_t = \Sigma_{t|t-1} \left( \Sigma_{t|t-1} + Q \right)^{-1}
  \]
  \[
  \Sigma_t = (I - K_t) \Sigma_{t|t-1}
  \]
  \[
  \mu_t = \mu_{t|t-1} + K_t y_t
  \]

Innovation (residual): \( y_t := z_t - \mu_{t-1} \)

\( z_t = x_t + N(0, Q) \)
Discussion

• Prediction: \( \mu_{t|t-1} = \mu_{t-1} \quad \Sigma_{t|t-1} = \Sigma_{t-1} + R \)

• Predicted measurement = predicted state

• Measurement update:

\[
\Sigma_t = (I - K_t) \Sigma_{t|t-1}
\]

\[
\mu_t = \mu_{t|t-1} + K_t y_t
\]

Kalman gain

What happens, if:

- \(R \to \infty\)
- \(Q \to \infty\)
- \(\Sigma \to 0\)

\[
y_t := z_t - \mu_{t-1} \text{ Innovation (residual)}
\]

\[
z_t = x_t + N(0, Q)
\]
Discussion

• What happens, if the object motion violates our model assumption
• Effects?
  – Big innovations, tracking lags
• Possible solutions?
  – Adapt noise settings (increase process noise/reduce meas. noise)
  – Choose a better model
We can estimate velocity!

past measurements

prediction
Simple constant velocity model

- Motion/dynamic model:
  \[
  x_{t|t-1} = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix} + N(0, R)
  \]
  \[
  R = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_\varepsilon^2 & 0 \\ 0 & 0 & 0 & \sigma_\varepsilon^2 \end{pmatrix}
  \]

- Measurement model:
  \[
  \begin{pmatrix} x_{obs} \\ y_{obs} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix} + N(0, Q)
  \]
  \[
  Q = \begin{pmatrix} \sigma_\delta^2 & 0 \\ 0 & \sigma_\delta^2 \end{pmatrix}
  \]
Constant velocity model

Assuming that the object follows a const. velocity model:

😊 Better prediction
😊 Cont’d estimate, even if we don’t have measurements
Discussion

- Constant position model
  \[ x_t = \begin{pmatrix} x \\ y \end{pmatrix} \]

- Constant velocity model
  \[ x_t = \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix} \]
Discussion

How to track very agile motions?

• Increased process noise
• Constant acceleration model
  😊 More reactive tracking
  😞 Instable, if no measurements available

• Alternative solutions:
  – Switched model
  – Knowledge from airplane available?
Kalman limitations

- Uni-modal distributions fail for unpredicted motion
- Problem, if image processing depends on prediction (tracking vs. detection)

Possible solutions:
- Switched model (ground contact as control input)
- Multiple hypothesis tracking

Prediction too far from actual location to recover
Kalman filter limitations

- Applies to linear Gaussian models
- Many visual tracking problems are nonlinear, e.g. as soon as we move to tracking in 3D

Courtesy of G. Panin
Humanoid robot catches flying balls

- Several cameras observe flying balls
- The 3D trajectory is estimated
- A robot is supposed to catch the balls ➔ nonlinear estimation problem

Courtesy of U. Frese
Extended Kalman filter

Slides based on S. Thrun
Nonlinear models, Gaussian noise

- **Motion model:**
  \[ x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t \Rightarrow g(x_{t-1}, u_t, \varepsilon_t) \]

- **Measurement model:**
  \[ z_t = C_t x_t + \delta_t \Rightarrow z_t = h(x_t, \delta_t) \]

**Nonlinear function**
- (most general model)
- Often used: model with additive noise

\[ \varepsilon_t \sim N(0, R_t) \]
\[ \delta_t \sim N(0, Q_t) \]
\[ X \sim N(\mu, \Sigma) \]
\[ Y = AX + B \]
\[ \Rightarrow Y \sim N(A\mu + B, A\Sigma A^T) \]
Nonlinear function

- PDF obtained from 500,000 Monte Carlo samples + histogramming
- Then sample mean and covariance
• **Problem:**
  We do not stay in the Gaussian world, if motion and/or measurement models are nonlinear functions of the state

\[
bel(x_t) = \eta p(z_t | x_t) \int p(x_t | x_{t-1}, u_t) \; bel(x_{t-1}) \; dx_{t-1}
\]

- No general closed-form solution for Bayes filter
- **Approximate solutions:**
  - Keep the functions and approximate distributions
  - **Linearize the functions and use again the Kalman filter**
Monte Carlo transform + sample statistics $\rightarrow$ blue Gaussian
First Order Taylor approximation $\rightarrow$ red Gaussian

Mismatch $= \text{linearization error}$
EKF linearization: additive noise

- Linearize $g()$ and $h()$ with **First Order Taylor Expansion**
- Linearization points: best available estimate
- Prediction (linearize around $\mu_{t-1}$):
  \[
  g(u_t, x_{t-1}) \approx g(u_t, \mu_{t-1}) + \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}} (x_{t-1} - \mu_{t-1})
  \]
  \[
  g(u_t, x_{t-1}) \approx g(u_t, \mu_{t-1}) + G_t (x_{t-1} - \mu_{t-1})
  \]
  
  **Jacobian matrix**

- Correction (linearize around $\bar{\mu}_t$):
  \[
  h(x_t) \approx h(\bar{\mu}_t) + \frac{\partial h(\bar{\mu}_t)}{\partial x_t} (x_t - \bar{\mu}_t)
  \]
  \[
  h(x_t) \approx h(\bar{\mu}_t) + H_t (x_t - \bar{\mu}_t)
  \]
  
  **Jacobian matrix**
Extended Kalman filter: additive noise

1. **Extended_Kalman_filter**\((\mu_{t-1}, \Sigma_{t-1}, u, z_t)\):

2. **Prediction**:

3. \(\bar{\mu}_t = g(u_t, \mu_{t-1})\) \hspace{2cm} \(\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t\)

4. \(\bar{\Sigma}_t = G_t \Sigma_{t-1} G_T^T + R_t\) \hspace{2cm} \(\bar{\Sigma}_t = A_t \Sigma_{t-1} A_T^T + R_t\)

5. **Correction**:

6. \(K_t = \bar{\Sigma}_t H_T^T (H_T \bar{\Sigma}_t H_T^T + Q_t)^{-1}\) \hspace{2cm} \(K_t = \bar{\Sigma}_t C_T (C_T \bar{\Sigma}_t C_T^T + Q_t)^{-1}\)

7. \(\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t))\) \hspace{2cm} \(\mu_t = \bar{\mu}_t + K_t (z_t - C_t \mu_t)\)

8. \(\Sigma_t = (I - K_t H_T) \bar{\Sigma}_t\) \hspace{2cm} \(\Sigma_t = (I - K_t C_T) \bar{\Sigma}_t\)

9. **Return** \(\mu_t, \Sigma_t\)

\[
H_t = \frac{\partial h(\bar{\mu}_t)}{\partial x_t} \quad G_t = \frac{\partial g(u_t, \mu_{t-1})}{\partial x_{t-1}}
\]
Linearization: non-additive noise

- Linearization points: best estimate and 0 (assuming zero-mean noise)
- Prediction (linearize around $\mu_{t-1}$ and 0):

$$g(u_t, x_{t-1}, \varepsilon_t) \approx g(u_t, \mu_{t-1}, 0) + \frac{\partial g(u_t, \mu_{t-1}, 0)}{\partial x_{t-1}} (x_{t-1} - \mu_{t-1}) + \frac{\partial g(u_t, \mu_{t-1}, \varepsilon_t)}{\partial \varepsilon_t} \varepsilon_t$$

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

- Correction (linearize around $\bar{\mu}_t$ and 0):

$$h(x_t, \delta_t) \approx h(\bar{\mu}_t, 0) + \frac{\partial h(\bar{\mu}_t, 0)}{\partial x_t} (x_t - \bar{\mu}_t) + \frac{\partial h(\bar{\mu}_t, \delta_t)}{\partial \delta_t} \delta_t$$

$$h(x_t, \delta_t) \approx h(\bar{\mu}_t, 0) + H_t (x_t - \bar{\mu}_t) + V_t \delta_t$$
Extended Kalman filter: non-additive noise

1. **Extended_Kalman_filter**($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

2. **Prediction**: 
   
3. $\bar{\mu}_t = g(u_t, \mu_{t-1})$
4. $\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + W_t R_t W_t^T$
5. **Correction**: 
6. $K_t = \bar{\Sigma}_t H_t^T (H_t \bar{\Sigma}_t H_t^T + V_t Q_t V_t^T)^{-1}$
7. $\mu_t = \bar{\mu}_t + K_t (z_t - h(\bar{\mu}_t))$
8. $\Sigma_t = (I - K_t H_t) \bar{\Sigma}_t$
9. **Return** $\mu_t, \Sigma_t$

\[ G_t = \frac{\partial g(u_t, \mu_{t-1}, 0)}{\partial x_{t-1}} \]
\[ W_t = \frac{\partial g(u_t, \mu_{t-1}, \varepsilon_t)}{\partial \varepsilon_t} \]
\[ H_t = \frac{\partial h(\mu_t, 0)}{\partial x_t} \]
\[ V_t = \frac{\partial h(\mu_t, \delta)}{\partial \delta_t} \]
Let:

- \( p(x) \) be unknown with mean \( \mu_x \) and covariance \( \Sigma_x \)
- \( y = f(x) \), a nonlinear function of \( x \)

The Gauss approximation for mean \( \mu_y \) and covariance \( \Sigma_y \) is:

\[
\mu_y = E(y) = E(f(x)) \approx E\left(f(\mu_x) + \frac{\partial f(\mu_x)}{\partial x}(x - \mu_x)\right) \\
= E\left(f(\mu_x)\right) + \frac{\partial f(\mu_x)}{\partial x}E(x - \mu_x) = f(\mu_x)
\]

\[
\Sigma_y = \text{cov}(y) = \text{cov}(f(x)) = E\left((f(x) - \mu_y)(f(x) - \mu_y)^T\right) \\
\approx E\left(\left(f(\mu_x) + \frac{\partial f(\mu_x)}{\partial x}(x - \mu_x) - f(\mu_x)\right)(\mu_y - \mu_y)^T\right) \\
= \frac{\partial f(\mu_x)}{\partial x}E\left((x - \mu_x)(x - \mu_x)^T\right) \frac{\partial f(\mu_x)^T}{\partial x}
\]
EKF linearization: problems

- Bigger uncertainty → bigger linearization error
EKF linearization: problems
EKF linearization: problems

• Higher local nonlinearity $\rightarrow$ bigger linearization error
EKF linearization: problems
Extended Kalman filter: summary

- Kalman Filter was optimal for linear Gaussian models
- Problem with EKF: the functions $g()$ and $h()$ are linearized, hence approximated $\Rightarrow$ Not optimal (we keep Gaussian representations, but $\text{bel}(x)$ is not Gaussian)
- If $g()$ and $h()$ are highly nonlinear (difficult to quantize), the estimation may become unstable
- Good results, if functions are approx. linear around mean
- Less certain estimate is more affected by linearization errors

**EKF is not an optimal Bayesian tracker**

- However, it is successfully used in many applications
(Extended) Kalman filter: limitations

- Question: is the (E)KF feasible for the above estimation problem?
- Implications of using Gaussians:
  - Gaussians are unimodal! They possess a single maximum!
  - Typical for many tracking problems: posterior focused around true state with small margin of uncertainty
  - Poor match for multimodal problem (global estimation)
(Extended) Kalman filter: limitations

No non-Gaussian observation models

M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. Van Gool, Robust Tracking-by-Detection using a Detector Confidence Particle Filter, International Conference on Computer Vision (ICCV), 2009
More Bayes filters (for the most interested ones)

- Iterated Extended Kalman filter
- Unscented Kalman filter
- Information filter

- Histogram filter
- Particle filter

- Interacting multiple model filter
- Multiple hypothesis tracking

Alternatives for (E)KF

- Nonlinear functions, multiple modes, non-Gaussian noise (in general less efficient)

- Mixture of Gaussians, fixed number of modes, different model assumptions

Bibliography: Books by Sebastian Thrun and Y. Bar-Shalom on http://av.dfki.de
Outlook

Next lectures:
• State-space models for 3D visual tracking and applications
• People tracking based on RGB-D data

Oral exam:
• Suggested periods:
  – 14. - 17.08.2012
  – (27. - 29.08.2012)
• Please write an email to leivy_michelly.kaul@dfki.de with cc to gabriele.bleser@dfki.de and ask for an appointment. The email should contain:
  – Full name, matriculation number, faculty, course of studies
  – Preferred date(s) and time