DS3 Göttingen 2019

August 5th – 16th 2019

Statistical Sensor Data and Information Fusion

Prof. Dr.-Ing. Marcus Baum

Fabian Sigges

Lecturer: Marcus Baum

Room: 3.115, Institut für Informatik
Tel: +49 551 39 172024
E-Mail: marcus.baum@cs.uni-goettingen.de

Since 2019	**Professor of Computer Science**
2015 - 2019	**Juniorprofessor**

GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN

2017 - 2019	**Visiting Professor**, Chair of Sensor Technology

UNIVERSITÄT PASSAU

2013/14	**Postdoc / Assistant Research Prof.** Peter Willett, Yaakov Bar-Shalom

UCONN UNIVERSITY OF CONNECTICUT

2013	**Dr.-Ing.,** Institute of Anthropomatics and Robotics
2007	**Dipl.-Inform.**

KIT Karlsruhe Institute of Technology

Tutor: Fabian Sigges

Room: 3.102, Institut für Informatik
Tel: +49 551 39 26039
E-Mail: fabian.sigges@cs.uni-goettingen.de

Since 2016	**Research Assistant** Institute of Computer Science	GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN
2016	**MSc Angwandte Informatik** Institute of Computer Science	GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN
2014	**BSc Angwandte Informatik** Institute of Computer Science	GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN

Data Fusion Lab

Head

- Marcus Baum

Assistance and Administration

- Tina Bockler
- Gunnar Krull

PhD Candidates

- Jaya Shradha Fowdur
  DLR, Institute of Communications and Navigation
- Hauke Kaulbersch
  Valeo Schalter und Sensoren GmbH
- Nils Kornfeld
  DLR, Institute of Transportation Systems
- Claudia Malzer
  Max-Planck-Institute for Dynamics and Self-Organization
- Simon Ollander
  Bosch Car & Multimedia, Hildesheim
- Fabian Sigges
- Kolja Thormann
- Shishan Yang

Schedule

Part 1: 14:00 - 15:30

- **Motivation and applications: Autonomous vehicles**
- **Theory: Kalman filter**

Part 2: 16:00 - 17:30

- **Exercises (in Python)**
- **Tutor: Fabian Sigges**

Motivation: Environmental Perception

Intelligent vehicles require map of the environment:

- Static objects
  (e.g., road boundaries)
- Moving objects
  (e.g., vehicles)

Sensor Fusion:

- Each Sensors has strengths and weaknesses

→ Combine different sensors

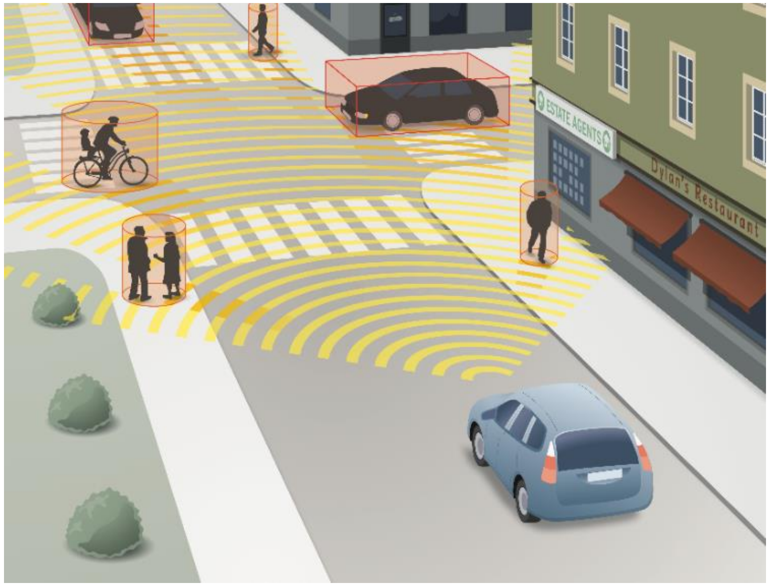


Illustration by Per Thorneus, Popill Agentur
Source: [Granström2012]

Source: https://www.youtube.com/watch?v=YtAwOl08dQQ

Sensors for Intelligent Vehicles

Radar

Smartmicro
Altimeter

- Measures velocity
- Weather and light independent
- Low resolution
- No color information

Camera

Mobileeye

- Color available
- High resolution and long range
- Weather and light dependent
- No velocity

Laser-Scanner

Valeo
SCALA

- Light independent
- High resolution and long range
- Weather dependent (snow/rain)
- No color information

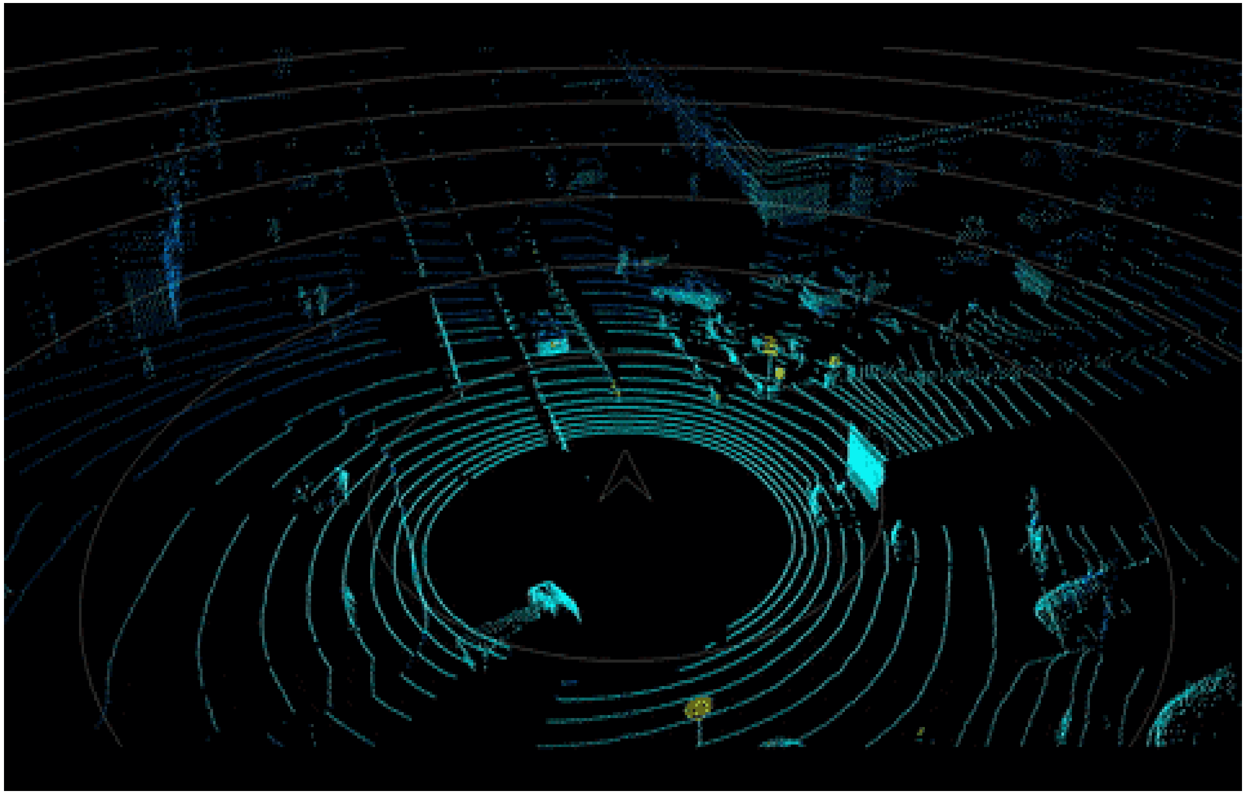
Ultra Sound

Valeo

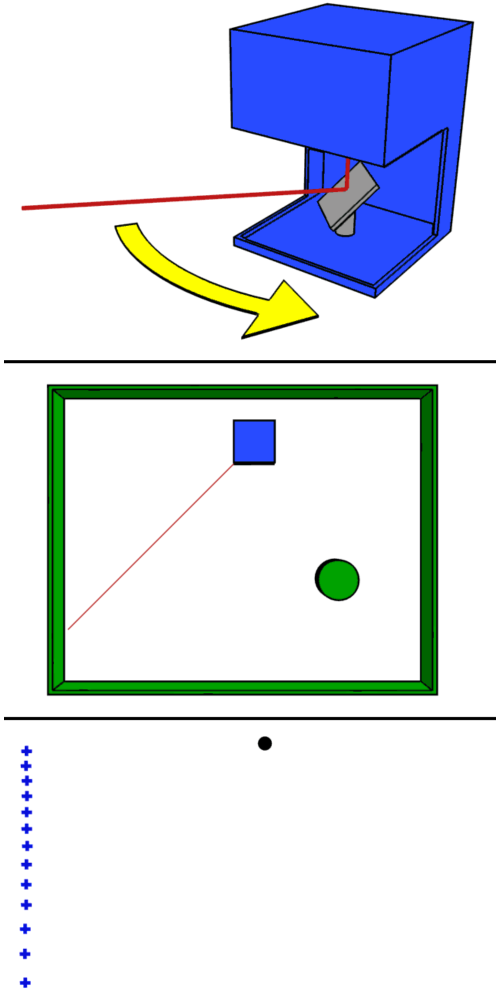
- Weather and light independent
- Small and cheap
- Low resolution and range
- No color information available

Laser Sensors (1)

LIDAR (**Li**ght **D**etection **A**nd **R**anging)

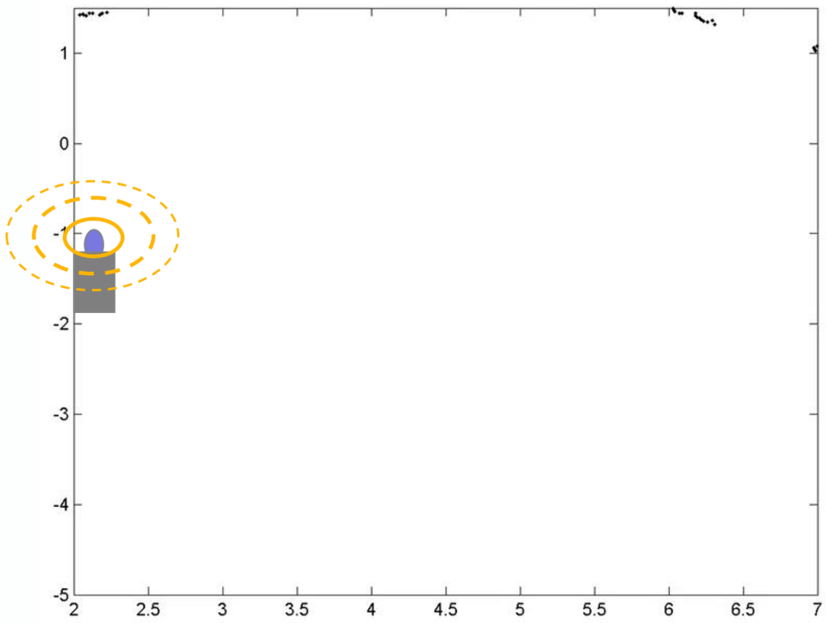


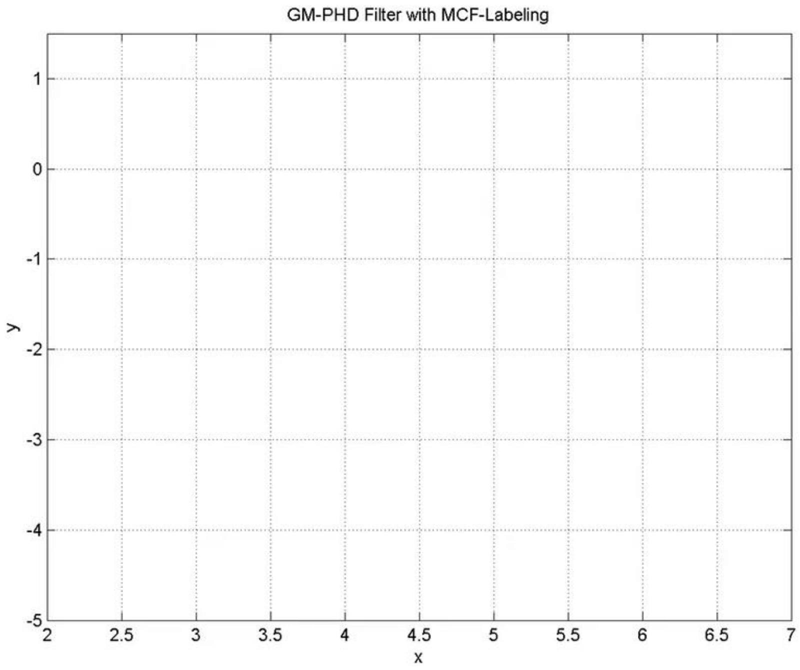
Source: HESAI, http://www.hesaitech.com/en/autonomous_driving.html



https://de.wikipedia.org/wiki/Lidar

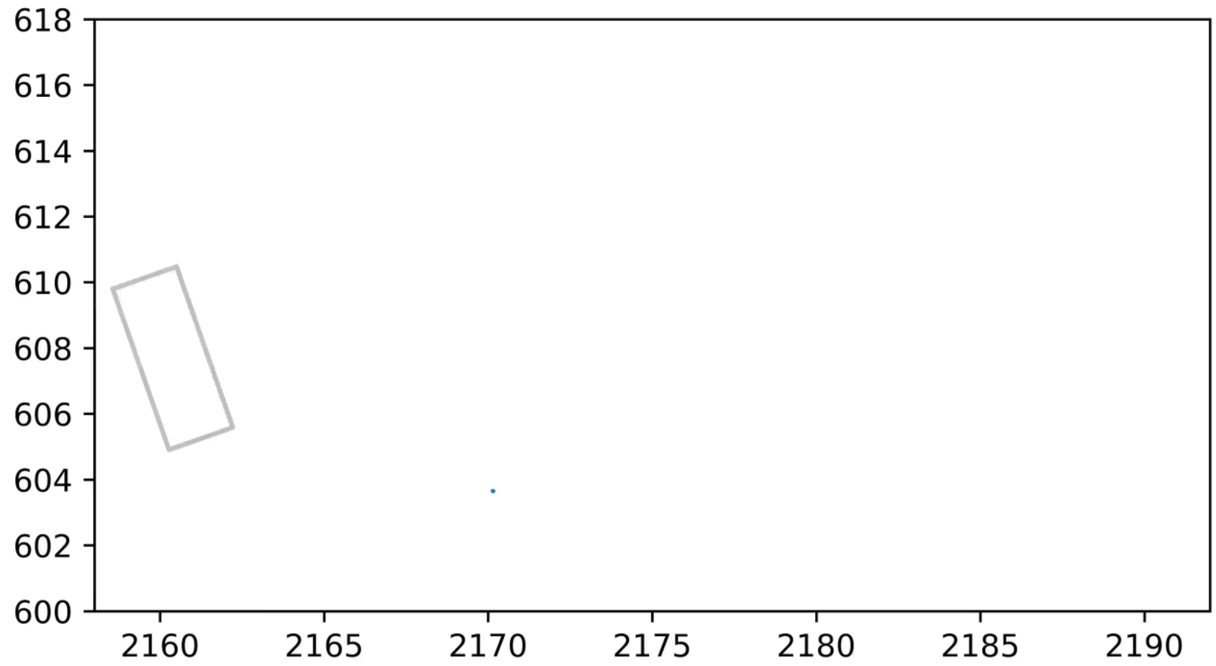
Laser Scanners (2)





Automotive Radar

- NuScenes dataset
  [Caesar et al, 2019]
- Extended object tracker
  [Kaulbersch et al, 2018]

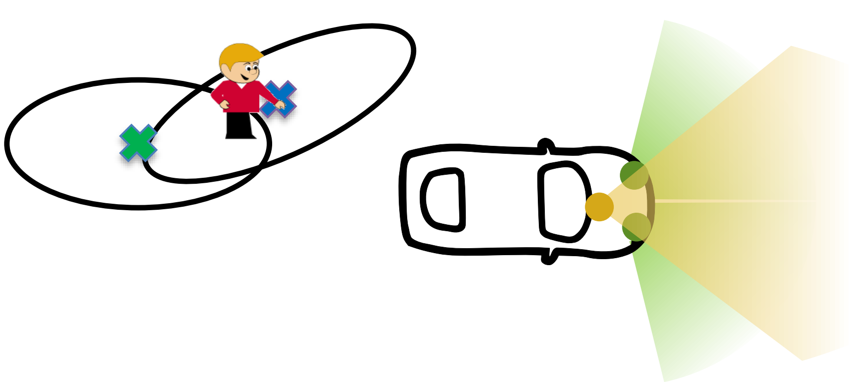


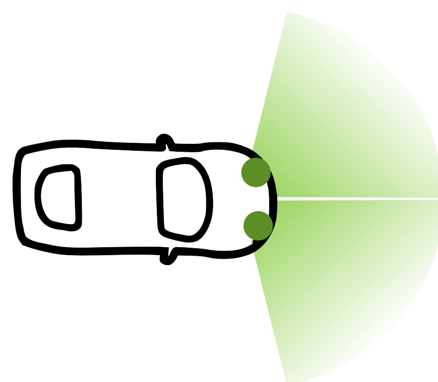
Sensorfusion: Concepts

Achievement: Reliability, Accuracy | Completeness

Fusion: **Competitive Fusion** | **Complementary Fusion**

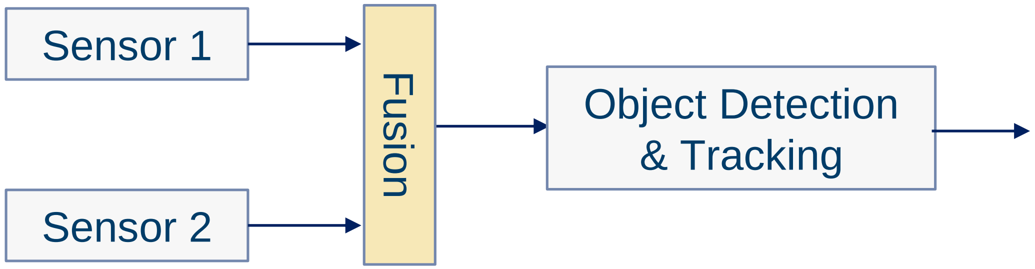
Sensors: S S | S S



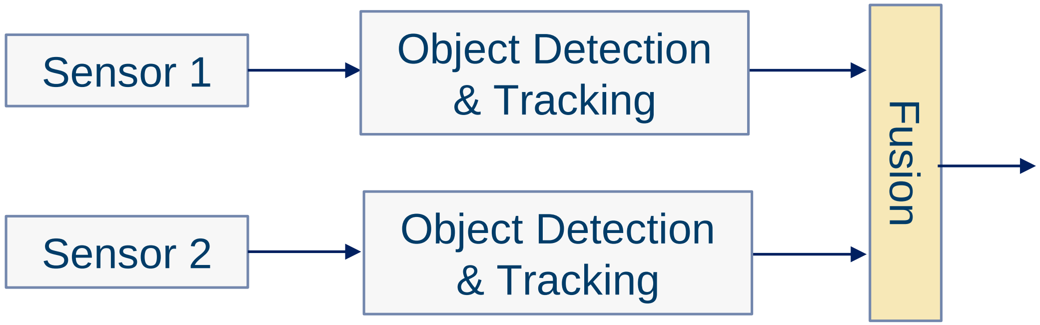


Fusion-Level

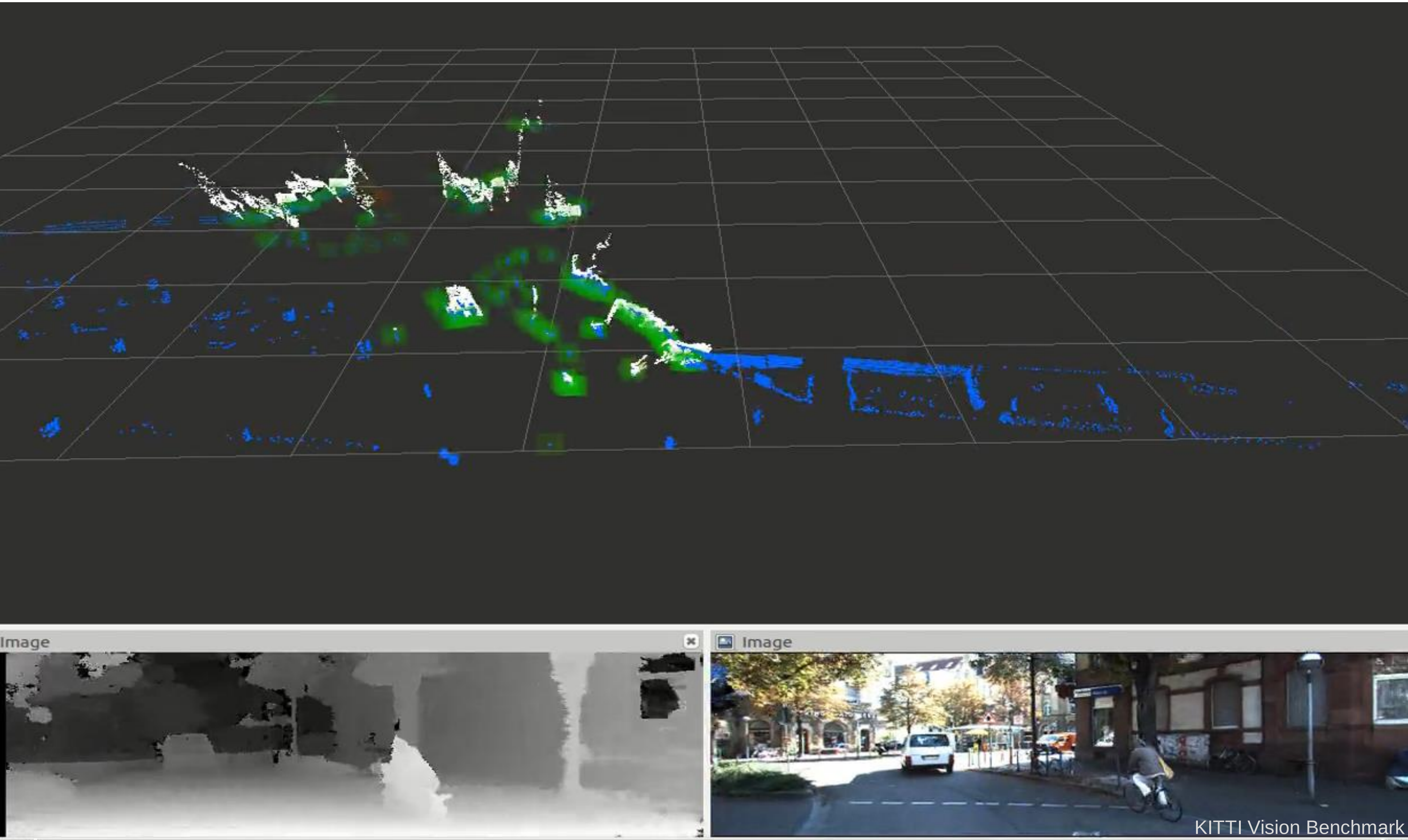
Low-Level-Fusion



Object-Level-Fusion



Low-Level-Fusion



Learning Outcome

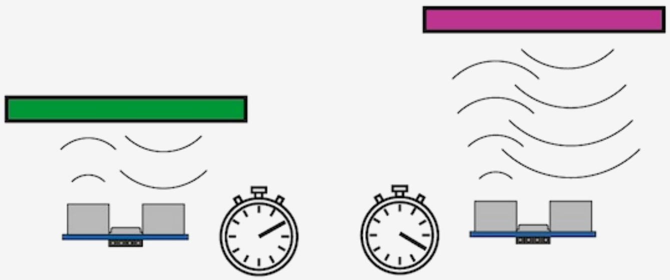
After the lecture you will be in the position to...

- fuse noisy measurements and state estimates,
- predict future states based on the current state estimate,
- explain the basic concept of the Kalman filter

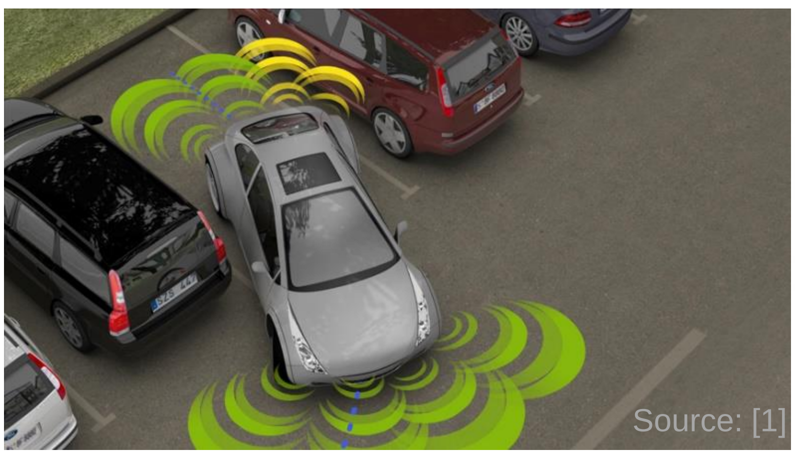
for the special case of scalar systems.

Ultrasonic Distance Sensor

- Speaker sends out sound waves
- Microphone measures time-of-flight
- Distance can be calculated based on the speed of sound

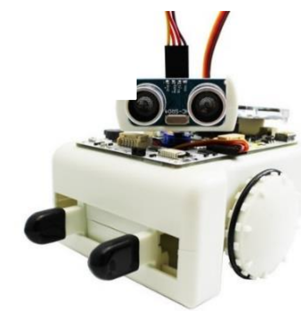


Source: [3]



Source: [1]

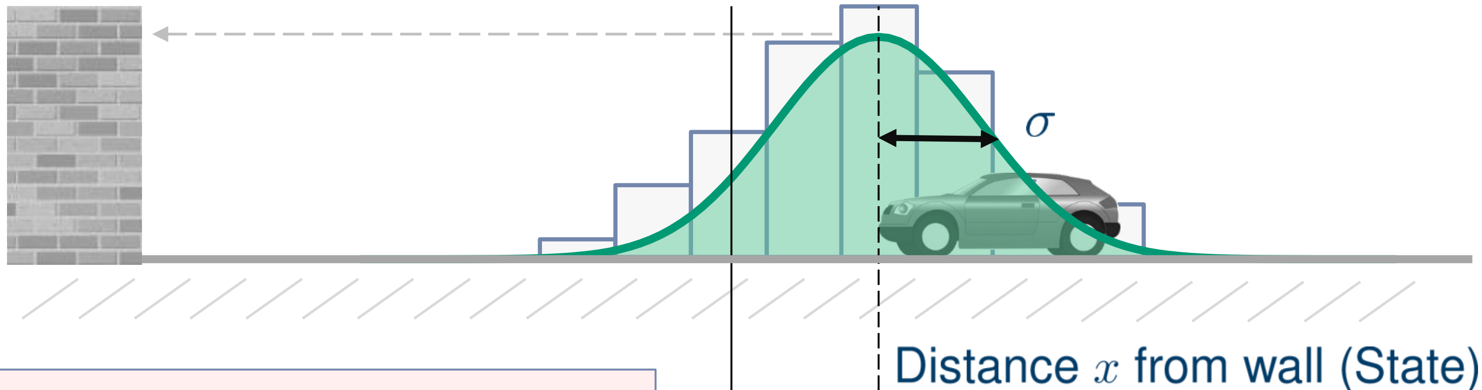
Source: [2]



Source: [3]

[1] https://www.valeo.de [2] https://www.parrot.com/de/drohnen/parrot-bebop-2 [3] http://arcbotics.com/products/sparki

Measurement Error



Measurements are always noisy

$\hat{y} = x + e$

$E(e^2)$

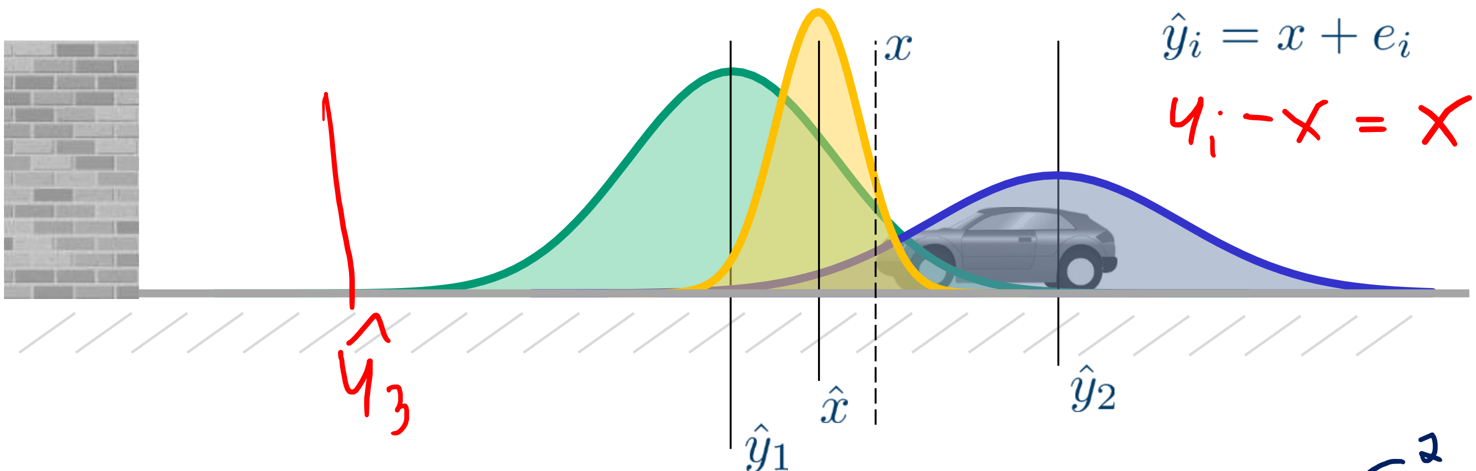
Measurement error e is (often) Gaussian distributed:

- Mean $\mathrm{E}[e] = 0$
- Variance $\mathrm{Var}[e] = \mathrm{E}[(e - \cancel{\mathrm{E}[e]})^2] = \mathrm{E}[e^2] = \sigma^2$
- larger variance $\rightarrow$ larger mean square error

$\Rightarrow$ A measurement $\hat{y}$ is unbiased, i.e., $\mathrm{E}[\hat{y} - x] = 0$

Key idea:
Combine, i.e., fuse, multiple measurements to reduce the error

Update: Fusion of Two Noisy Measurements



$\sigma_2^2 \to \infty$

$\sigma_2^2 = 0$

Meas. / Estimate	Error Covariance
$\underline{\hat{y}_1}$	$\underline{\sigma_1^2} = \mathrm{E}[\underline{e_1^2}] = \mathrm{E}[(y_1 - x)^2]$
$\underline{\hat{y}_2}$	$\underline{\sigma_2^2} = \mathrm{E}[e_2^2] = \mathrm{E}[(y_2 - x)^2]$
$\hat{x} = \underline{(1-\alpha)\hat{y}_1 + \alpha\hat{y}_2}$	$\sigma_x^2 = \mathrm{E}[(\hat{x} - x)^2] = (1-\alpha)\sigma_1^2$

$$\alpha = \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}$$

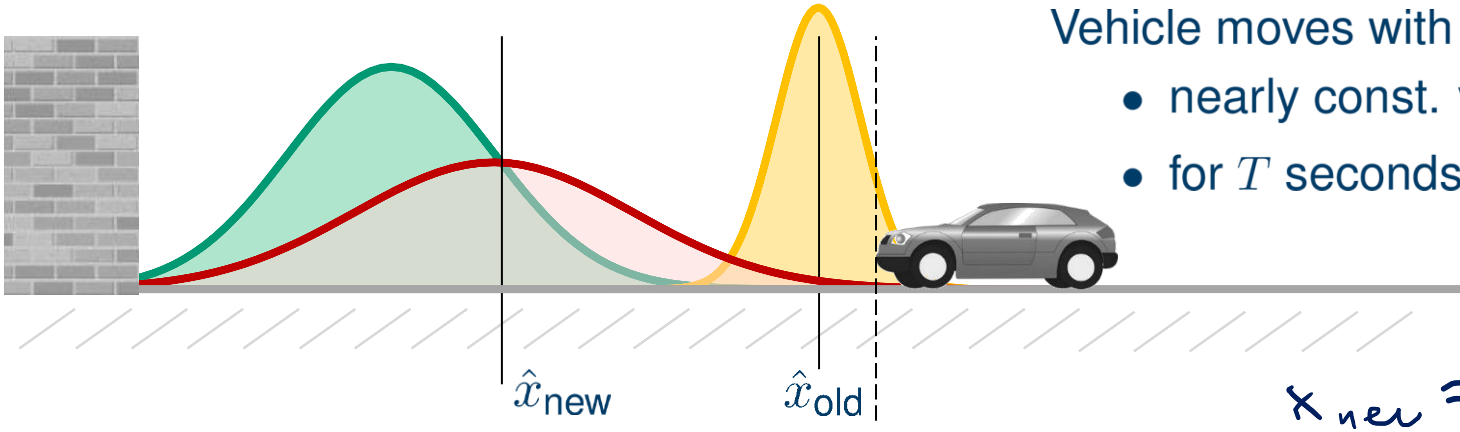
$\hat{y}_1$

Linear estimator

Variance decreases

Can be applied recursively

Time Update: Prediction of an Estimate



Vehicle moves with

- nearly const. velocity v
- for T seconds.

$x_{new} = \frac{1}{2} x_{old}$

$\hat{x}_{\text{old}}$	σ^2_{old}

Discrete-time Motion Model:

$$x_{\text{new}} = x_{\text{old}} + T \cdot (v + e_v)$$

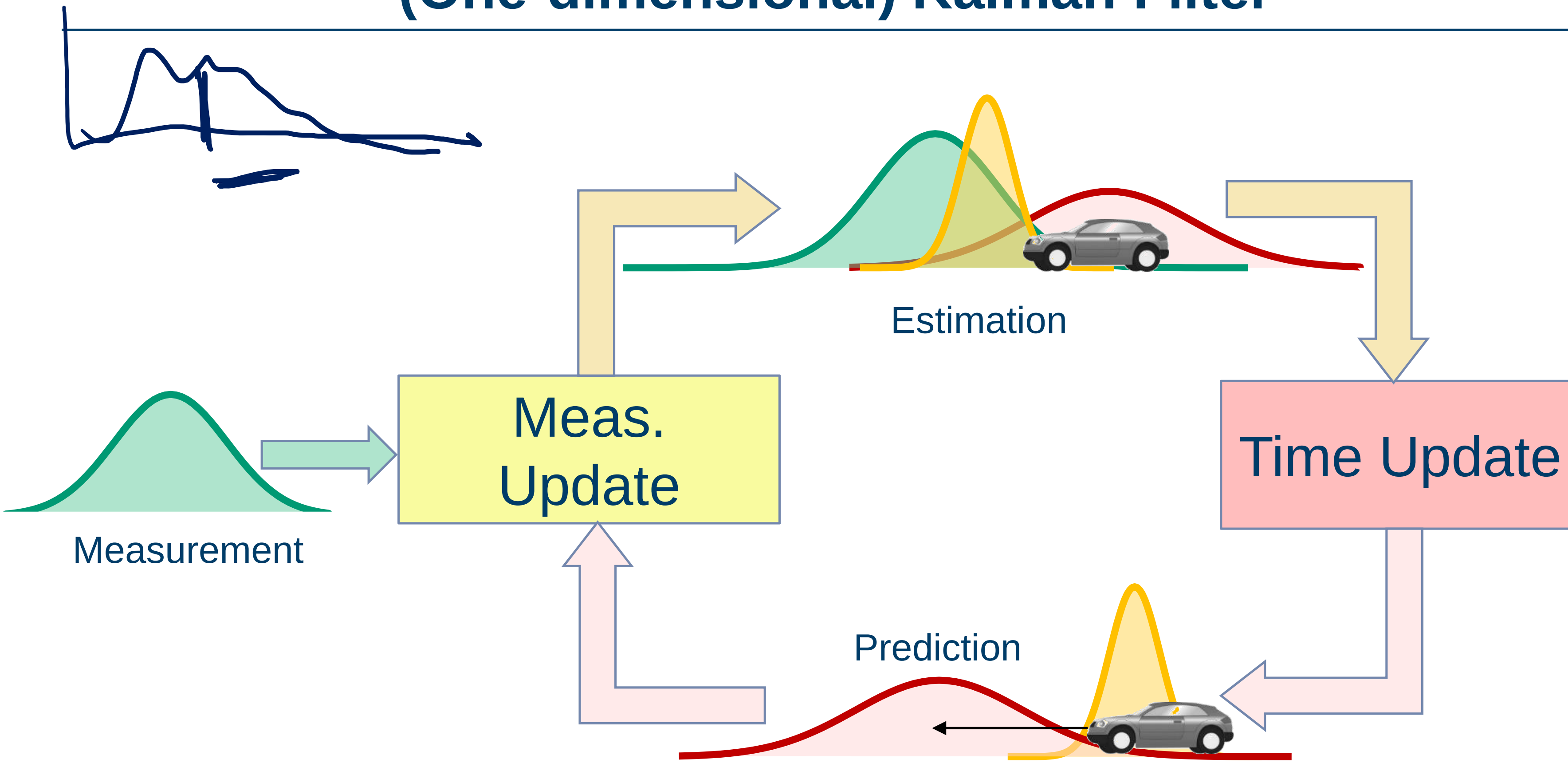
with velocity v and zero-mean noise e_v with variance σ_v^2

$\hat{x}_{\text{new}} = \hat{x}_{\text{old}} + T \cdot v$	$\sigma^2_{\text{new}} = \sigma^2_{\text{old}} + T^2 \sigma_v^2$

Variance increases

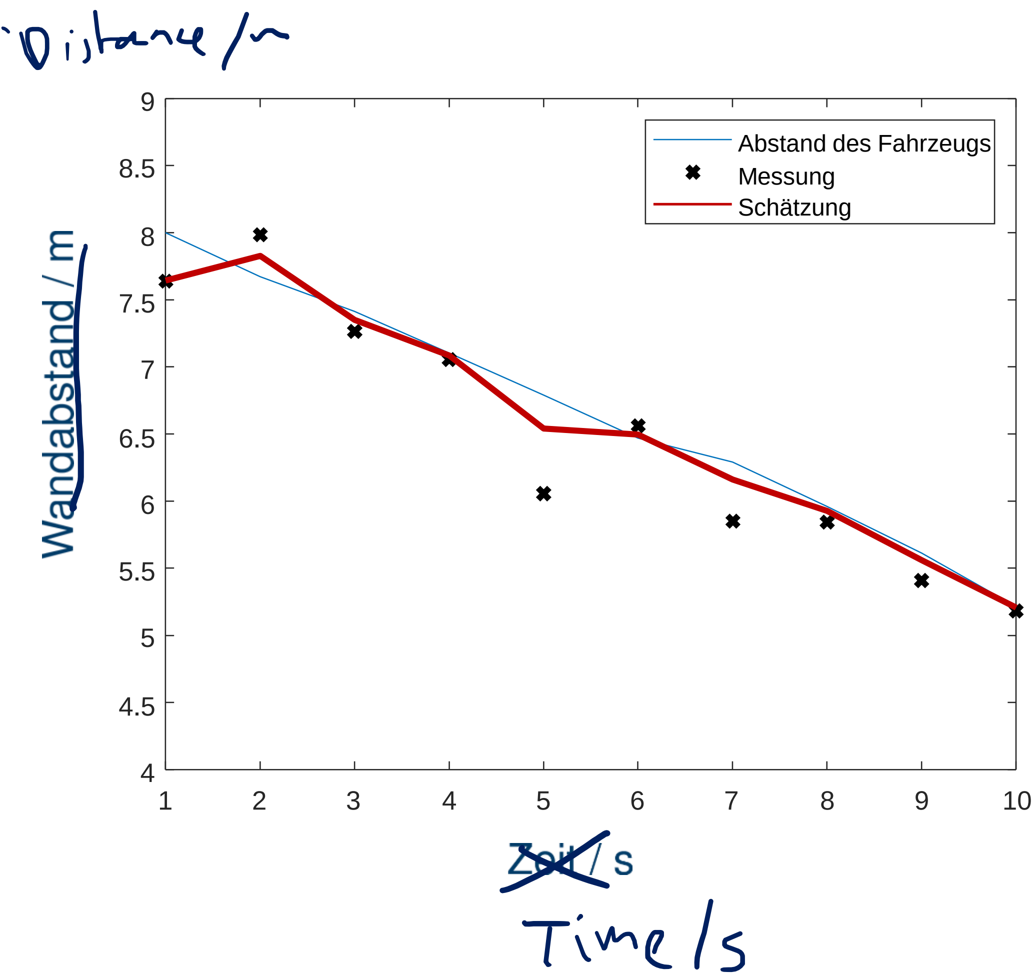
Next measurement: fusion with prediction

(One-dimensional) Kalman Filter



- Alternating measurement and time updates
- Recursive filter
- Optimal for linear systems with (white) Gaussian noise, otherwise best linear estimator

Numerical Example



- Time interval $T = 1\text{s}$
- Velocity $v = -0.3\,\frac{\text{m}}{\text{s}}$
- Meas. noise $\sigma_e = 0.3\text{m}$
- System noise $\sigma_v = 0.05\text{m}$

Derivation: Linear Transformation

Given is a scalar random variable x with $\mathrm{E}[x] = \mu_x$ and $\mathrm{Var}[x] = \sigma_x^2$

- Multiplication with a constant a:

$$\mathrm{E}[a \cdot x] = a\mu \text{ and } \mathrm{Var}[a \cdot x] = a^2\sigma_x^2$$

- Sum of x and constant b:

$$\mathrm{E}[x + b] = \mu + b \text{ and } \mathrm{Var}[x + b] = \sigma_x^2$$

- Sum of x and (independent) random variable z with $\mathrm{E}[z] = \mu_z$ and $\mathrm{Var}[z] = \sigma_z^2$:

$$\mathrm{E}[x + z] = \mu_x + \mu_z \text{ and } \mathrm{Var}[x + z] = \sigma_x^2 + \sigma_z^2$$

Derivation: Update Step

$$\hat{x} = (1-\alpha)\hat{y}_1 + \alpha\hat{y}_2 \qquad \sigma_x^2 = \mathrm{E}[(\hat{x}-x)^2] = (1-\alpha)\sigma_1^2$$

$$\alpha = \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2}$$

Unbiased:

- $\mathrm{E}[\hat{x}] = \mathrm{E}[(1-\alpha)(x+e_1) + \alpha(x+e_2)] = (1-\alpha)\,\mathrm{E}[x] + \alpha\,\mathrm{E}[x] = \mathrm{E}[x]$

Minimum Mean Squared Error, i.e., $\mathrm{Var}[\hat{x}-x] = \mathrm{E}[(\hat{x}-x)^2]$:

- For arbitrary β:
  $\mathrm{Var}[\hat{x}-x] = \mathrm{E}[(\hat{x}-x)^2] = \mathrm{Var}[(1-\beta)(x+e_1) + \beta(x+e_2) - x] =$
  $\mathrm{Var}[(1-\beta)e_1 + \beta e_2] = (1-\beta)^2\sigma_1^2 + \beta^2\sigma_2^2$
- Derivative w.r.t. β shall be zero:
  $-2(1-\beta)\sigma_1^2 + 2\beta\sigma_2^2 = 0 \;\rightarrow\; \sigma_1^2 = \beta(\sigma_1^2+\sigma_2^2) \;\rightarrow\; \beta = \alpha$
- $\mathrm{Var}[\hat{x}-x]$ results from substituting α

Derivation: Prediction Step

Discrete-time Motion Model:

$$x_{\text{new}} = x_{\text{old}} + T \cdot (v + e_v)$$

with speed v and zero-mean noise e_v with variance σ_v^2

$\hat{x}_{\text{new}} = \hat{x}_{\text{old}} + T \cdot v$	$\sigma_{\text{new}}^2 = \sigma_{\text{old}}^2 + T^2 \sigma_v^2$

Unbiased:

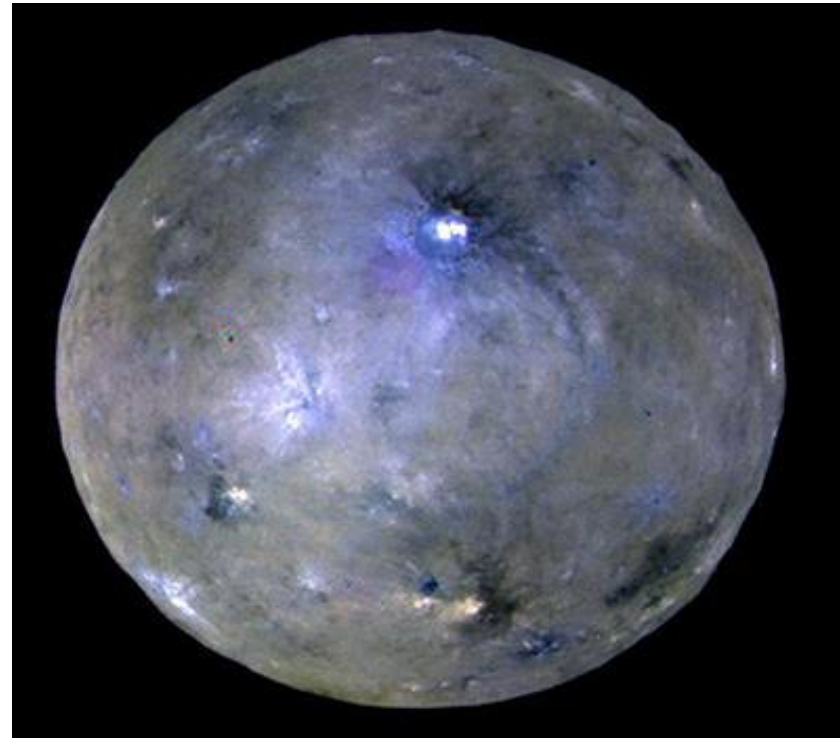
- $\mathrm{E}[\hat{x}_{\text{new}}] = \mathrm{E}[\hat{x}_{\text{old}} + T \cdot v] = \mathrm{E}[x_{\text{old}}] + T \cdot v = \mathrm{E}[x_{\text{new}}]$

Mean Squared Error:

- $\mathrm{E}[(x_{\text{new}} - \hat{x}_{\text{new}})^2] = \mathrm{E}[(x_{\text{old}} + T \cdot (v + e_v) - \hat{x}_{\text{old}} - T \cdot v)^2]$
  $= \sigma_{\text{old}}^2 + T^2 \sigma_v^2$

Discovery of Ceres

- On January 1, 1800,
  Giuseppe Piazzi, discovered Ceres in Palermo
- He recorded its position
- Over 42 days, Piazzi collected 19 measurements
- On February 12, the object disappeared
- Total motion covered an arc of 3°



Source:
https://solarsystem.nasa.gov/missions/dawn/science/ceres/

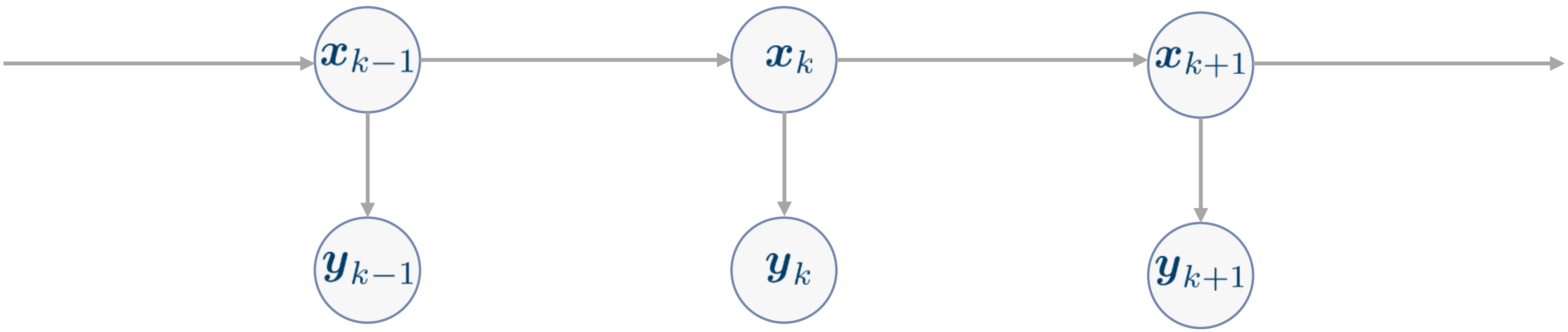
- C. F. Gauss (24 years)
  predicted the Ceres orbit in´1801
- Ceres was rediscovered by Franz X. von Zach on 31 December
- Gauss used the method of least squares (developed in 1795 with 18 years)

Rudolf E. Kalman (1930 - 2016)

- Born in Budapest,1930
- Emigrated to the US in 1943
- BA (1953) and MA (1954) from MIT
- PhD, 1957, Columbia University, NYC
- Professor at Stanford University (1964-1971), University of Florida (1971-1992) and ETH (since 1973)

- First publication of the Kalman Filter in 1960
- First implementation by Stanley F. Schmidt (NASA Ames Research Center)
- Used for trajectory estimation in the Apollo Program
- Many co-inventors and related works (e.g., Swerling, Bucy, Schmidt, Stratonovich, Gauss, …)

State Space Model



- State vector $\boldsymbol{x}_k \in \mathbb{R}^{m_x}$ for $k \in \mathbb{Z}$
- Observations/Measurements: $\boldsymbol{y}_k \in \mathbb{R}^{m_y}$

Linear Dynamic Systems (1)

- State vector at discrete time $k \in \mathbb{N}_0$:

$$x_k \in \mathbb{R}^n$$

  with initial mean $\mathrm{E}[x_0] = \hat{x}_0$ and covariance $\mathrm{Cov}[x_0] = \mathbf{C}_0^{xx}$

- Process model:

$$x_{k+1} = \mathbf{A}_k x_k + \mathbf{B}_k u_k + w_k$$

  - State transition matrix $\mathbf{A}_k \in \mathbb{R}^{n \times n}$
  - Control input $u_k \in \mathbb{R}^p$
  - Control input matrix $\mathbf{B}_k \in \mathbb{R}^{n \times p}$
  - Zero-mean white process noise $w_k \in \mathbb{R}^n$:
    * $\mathrm{E}[w_k] = \mathbf{0}_n$
    * $\mathrm{E}[w_k w_k^T] = \mathbf{C}_k^{ww}$
    * $\mathrm{E}[w_k w_l^T] = \mathbf{0}_{n \times n}$ for $k \neq l$

Linear Dynamic Systems (2)

- Measurement model:

$$y_k = \mathbf{H}_k x_k + v_k$$

  - Measurement matrix $\mathbf{H}_k \in \mathbb{R}^{m \times n}$
  - Zero-mean white measurement noise $v_k \in \mathbb{R}^m$ with
    * $\mathrm{E}[v_k] = \mathbf{0}_n$
    * $\mathrm{E}[v_k v_k^T] = \mathbf{C}_k^{vv}$
    * $\mathrm{E}[v_k v_l^T] = \mathbf{0}_{m \times m}$ for $k \neq l$
- Initial state, and all noise vectors mutually uncorrelated

Kalman Filter Formulas

- Recursively calculate estimate $\hat{x}_k$ and error covariance $\mathbf{C}_k^{xx}$ for time k
- Time update:

$$\begin{aligned} \hat{x}_{k+1|k} &= \mathbf{A}_k \hat{x}_k + \mathbf{B} u_k \\ \mathbf{C}_{k+1|k}^{xx} &= \mathbf{A}_k \mathbf{C}_k^{xx} \mathbf{A}_k^T + \mathbf{C}_k^{ww} \end{aligned}$$

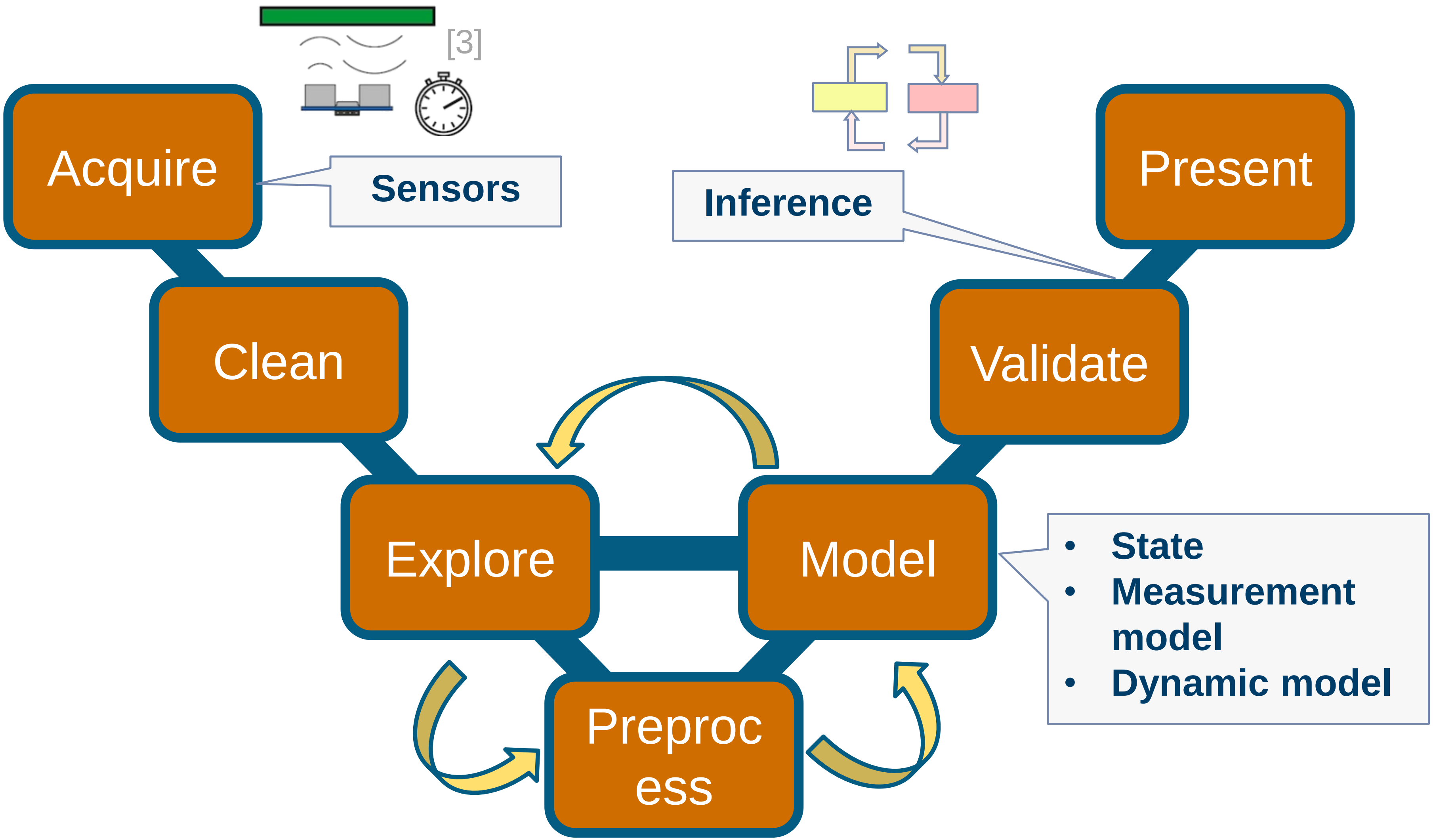
- Measurement update:

$$\begin{aligned} \hat{x}_{k+1} &= \hat{x}_{k+1|k} + \mathbf{K}_{k+1} \left(y_{k+1} - \mathbf{H} \hat{x}_{k+1|k} \right) \\ \mathbf{C}_{k+1}^{xx} &= \mathbf{C}_{k+1|k}^{xx} - \mathbf{K}_{k+1} \mathbf{H} \mathbf{C}_{k+1|k}^{xx} \end{aligned}$$

with Kalman gain

$$\mathbf{K}_{k+1} = \mathbf{C}_{k+1|k}^{xx} \mathbf{H}^T \left(\mathbf{H} \mathbf{C}_{k+1|k}^{xx} \mathbf{H}^T + \mathbf{C}_{k+1}^{vv} \right)^{-1}$$

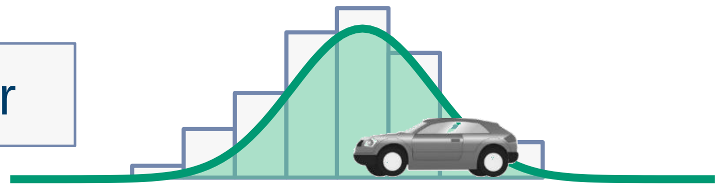
Data Science Building Blocks

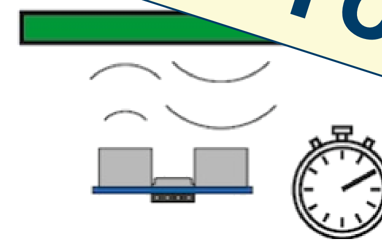


Summary

Thank You

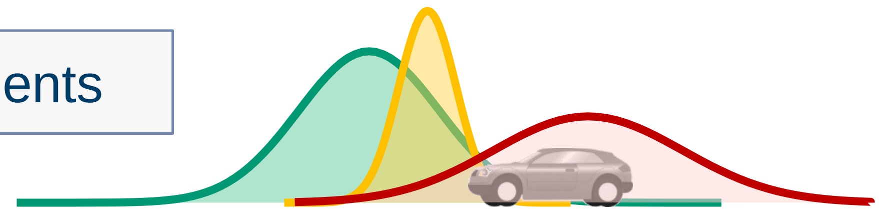
Measurement error



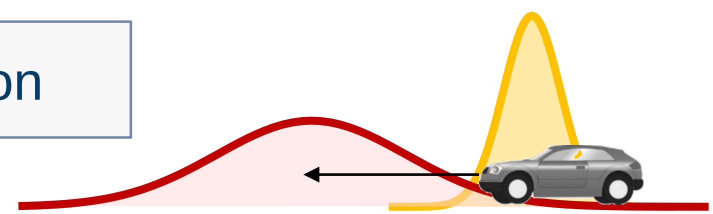


[3]

Fusion of measurements



Prediction



Time and meas. updates