1. Introduction to Multiple Object Tracking

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Outline

- Section 1: Introduction to Multiple Object Tracking
  1. Subsection: Motivating Examples
  2. Subsection: Definition of Multiple-Object-Tracking, types of tracking
  3. Subsection: Challenges in MOT
  4. Subsection: Prediction, update, likelihood
  5. Subsection: Brief Kalman filter review
  6. Subsection: Conjugate Prior definition

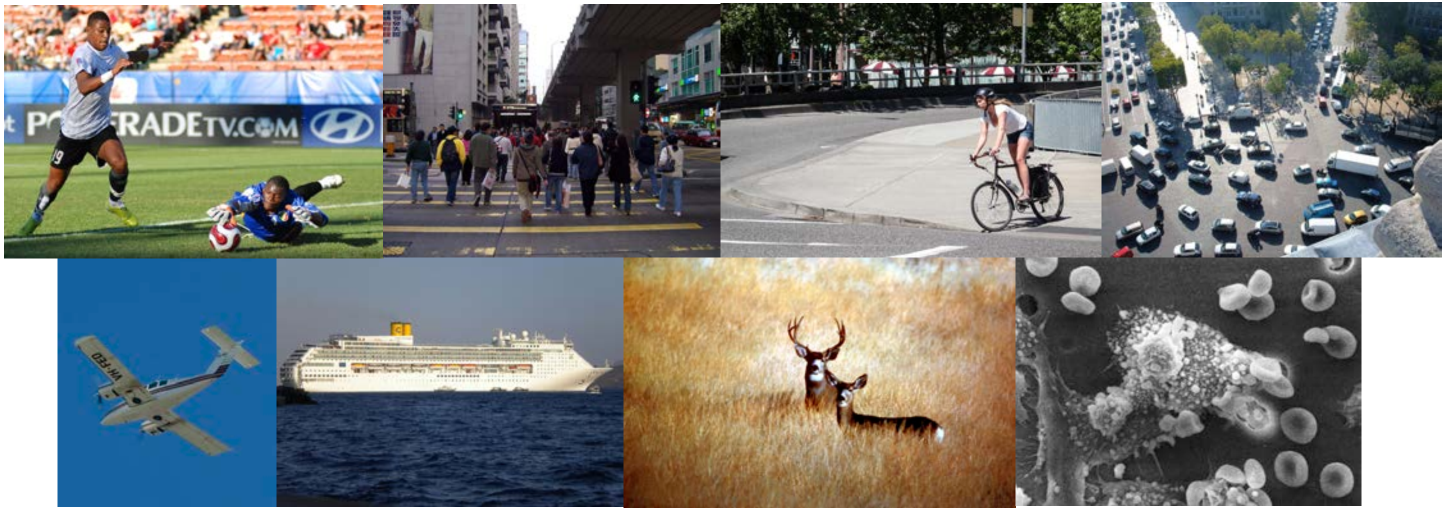
Karl Granström
Department of Electrical Engineering

1.1 Motivating examples

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Karl Granström
Department of Electrical Engineering

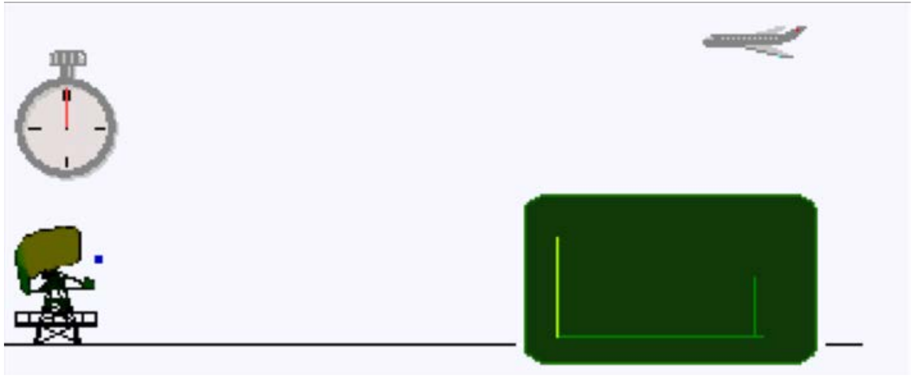
Moving objects of many different kinds



How many objects are there?
Where are they, and where are they going?
Important properties/characteristics?
Study this over time.

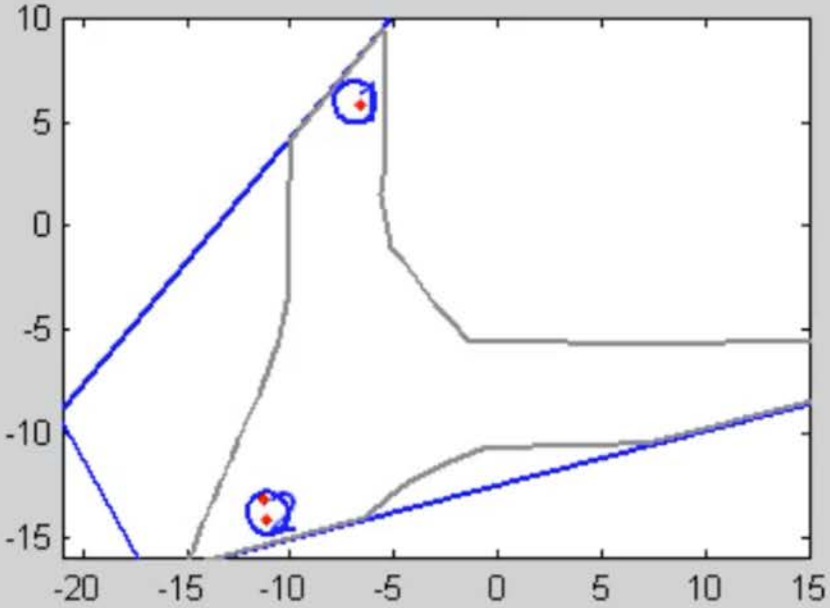
Images from Wikimedia Commons

Historical origins: Tracking airplanes using radar

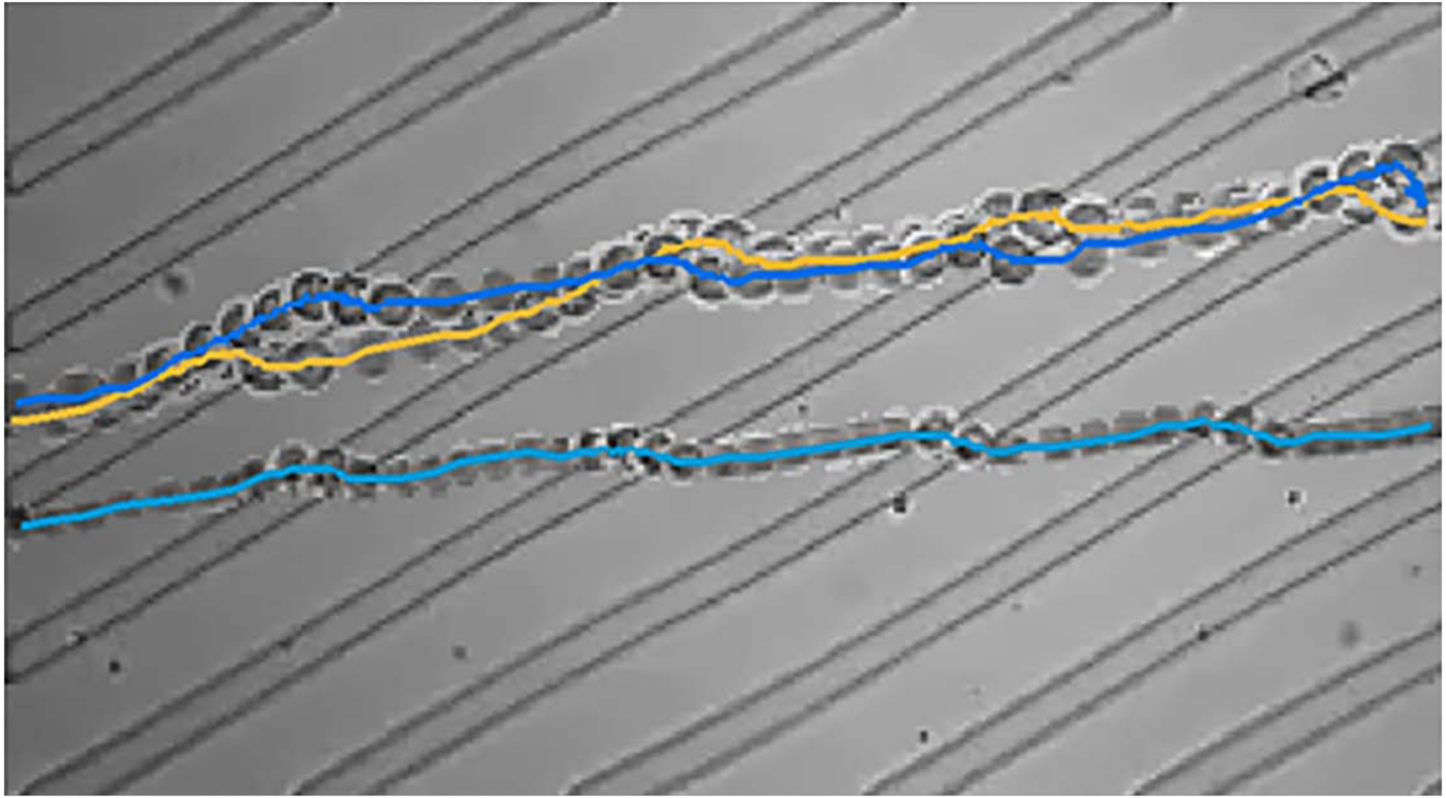


Airport surveillance using ground radar

Surveillance of groups of pedestrians

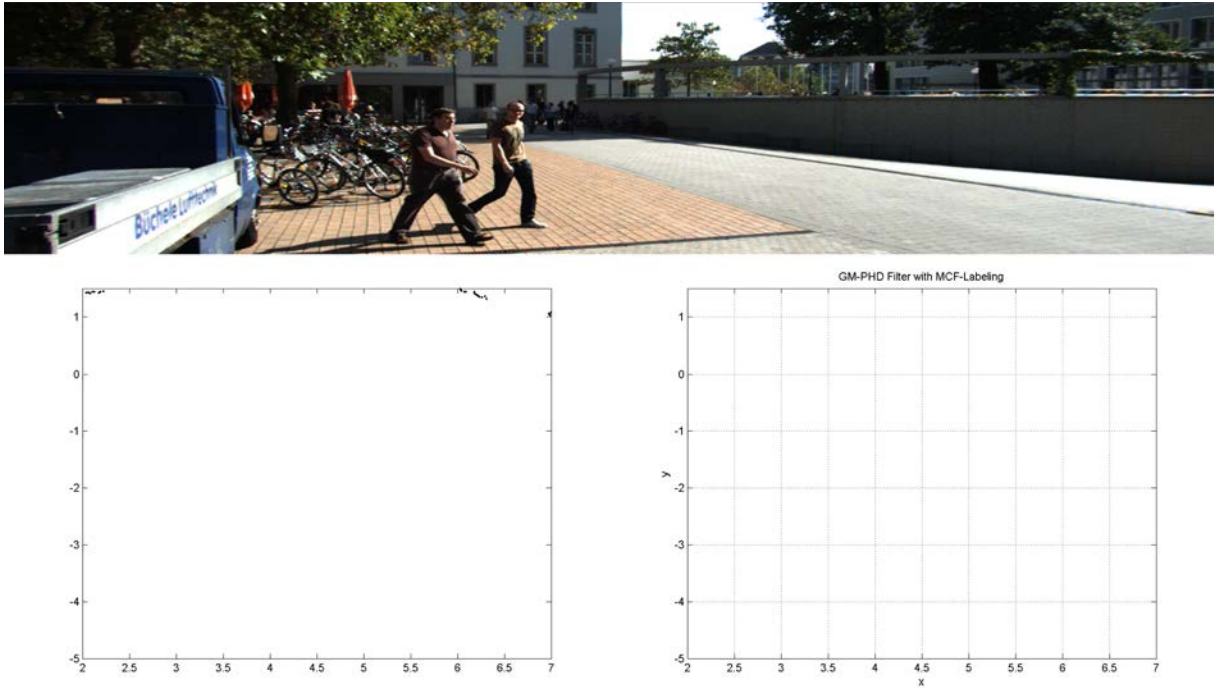


Microfluidic tracking to quantify the biomechanical properties of the cells



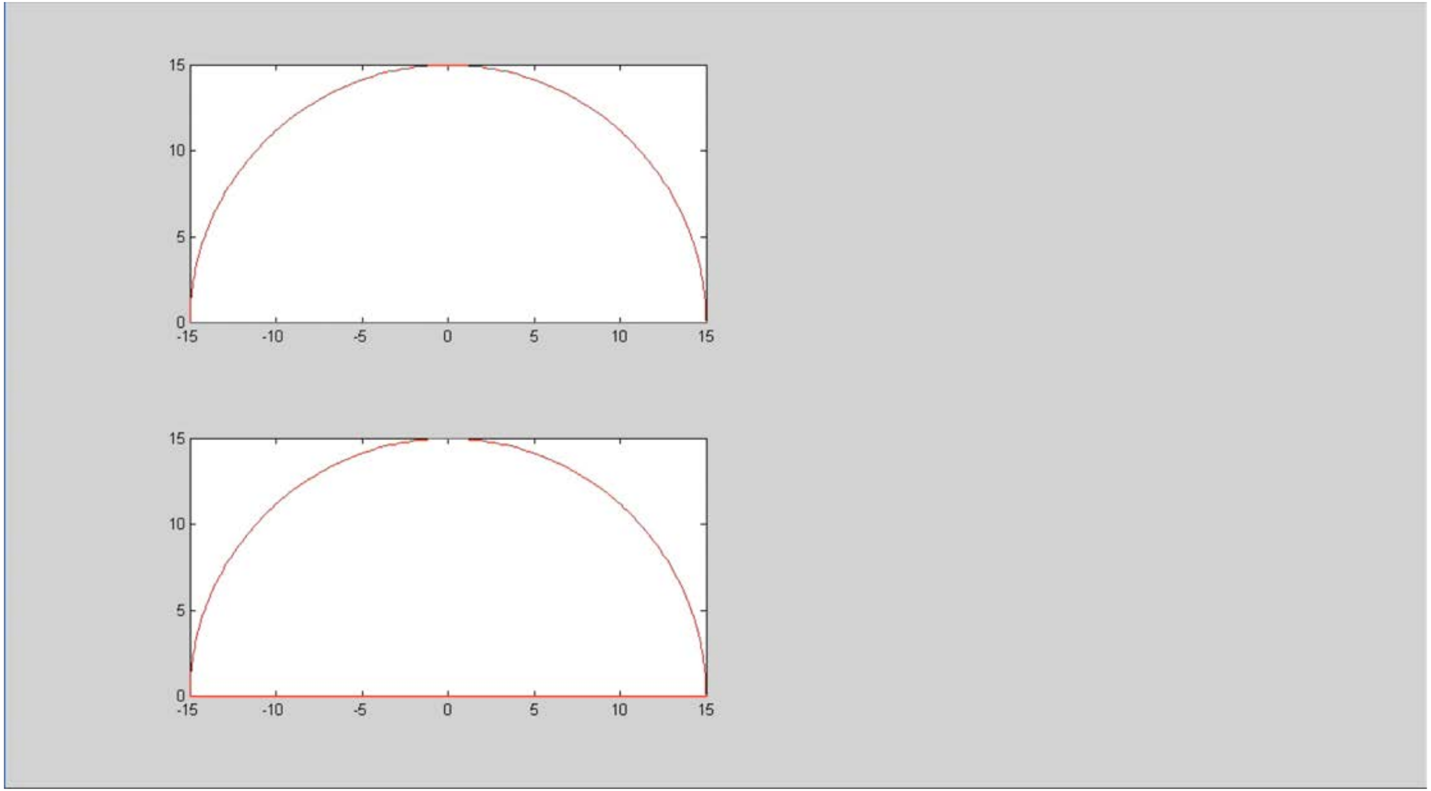
A short segment of video recording shown by overlapping multiple frames, and single-cell trajectories.
J. Jeong et al (2018), *Accurately tracking single-cell movement trajectories in micrefluidic cell sorting devices*, PLOS One, 13(2)

Pedestrian tracking using Lidar



Florian Teich (2017), "*Multiple Extended Object Tracking using the Multiplicative Error Shape Model and Network Flow Labeling*", Georg-August-Universität Göttingen, Department for Computer Science, Data Fusion Group.
The data is from the KITTI Vision Benchmark Suite, see Geiger et al (2013), *"Vision meets Robotics: The KITTI Dataset"*, International Journal of Robotics Research

Bicycle tracking using Lidar

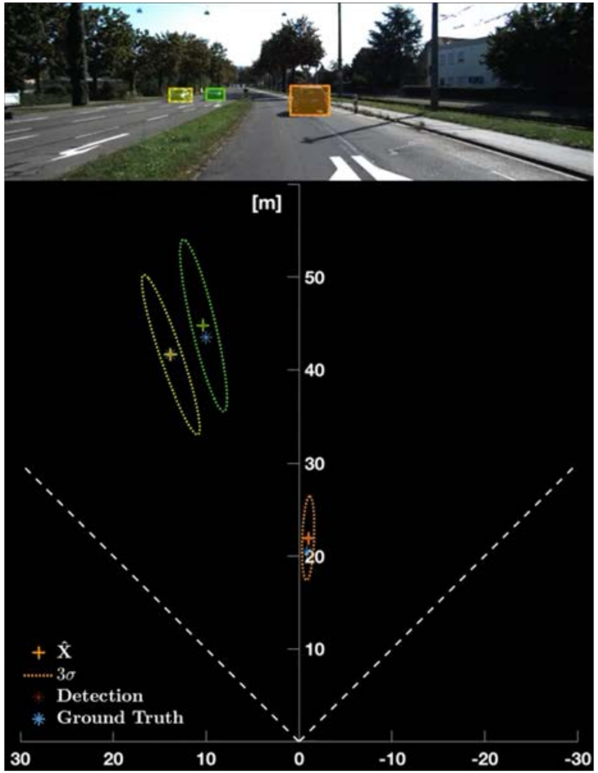


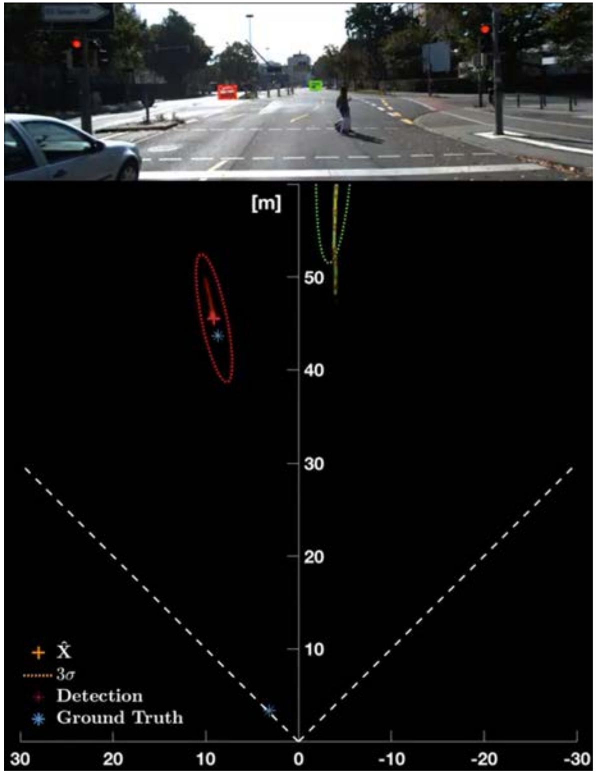
Granstrom & Lundquist (2013), "*On the use of multiple measurement models for extended target tracking"*, Proceedings of International Conference on Information Fusion

Vehicle tracking using automotive radar

Olafsson & Krishnamoorthy (2017), ``*Vehicle Trajectory Optimisation - Employing Extended Target Tracking, Guardrail Information and Smoothing Methods on Radar-only Detections"*, Master's Thesis, Department of Electrical Engineering, Chalmers University of Technology

Vehicle tracking using mono-camera data

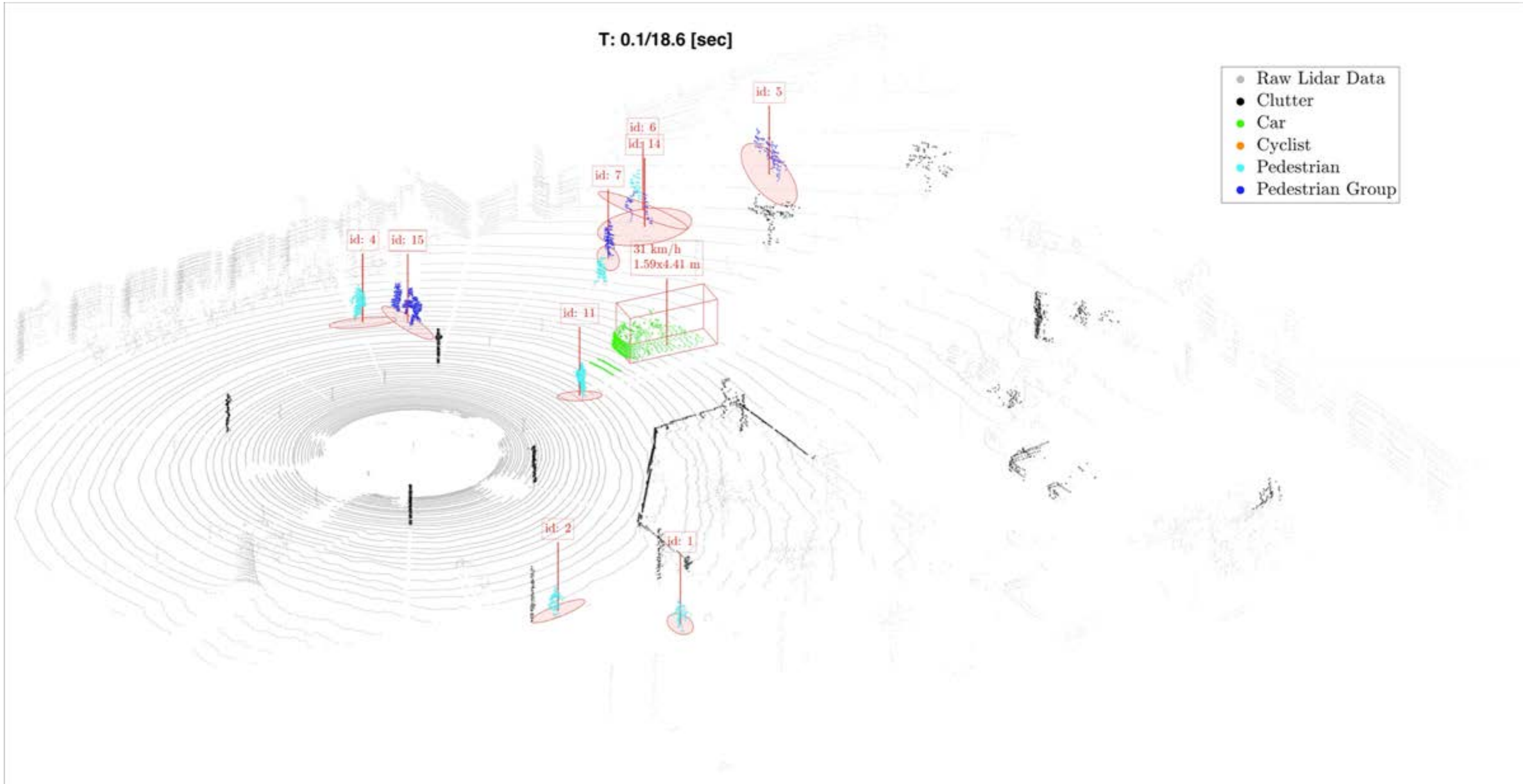




Scheidegger et al (2018), "*Mono-Camera 3D Multi-Object Tracking Using Deep Learning Detections and PMBM Filtering*", Proceedings of IEEE Intelligent Vehicles Symposium
More results at https://goo.gl/AoydgW

Data from KITTI Vision Benchmark Suite, see Geiger et al (2013), *"Vision meets Robotics: The KITTI Dataset"*, International Journal of Robotics Research

Tracking cars, bicyclists, and pedestrians using 3D lidar



Cotra & Leue (2016), *"Lidar-based Methods for Tracking and Identification"*, Master's Thesis, Department of Electrical Engineering, Chalmers University of Technology
https://youtu.be/_Mhgm2BXdFI

MOT applications

Tracking systems are a key technology for many technical applications in areas:

- robotics,
- surveillance,
- autonomous driving,
- automation,
- medicine, and
- sensor networks.

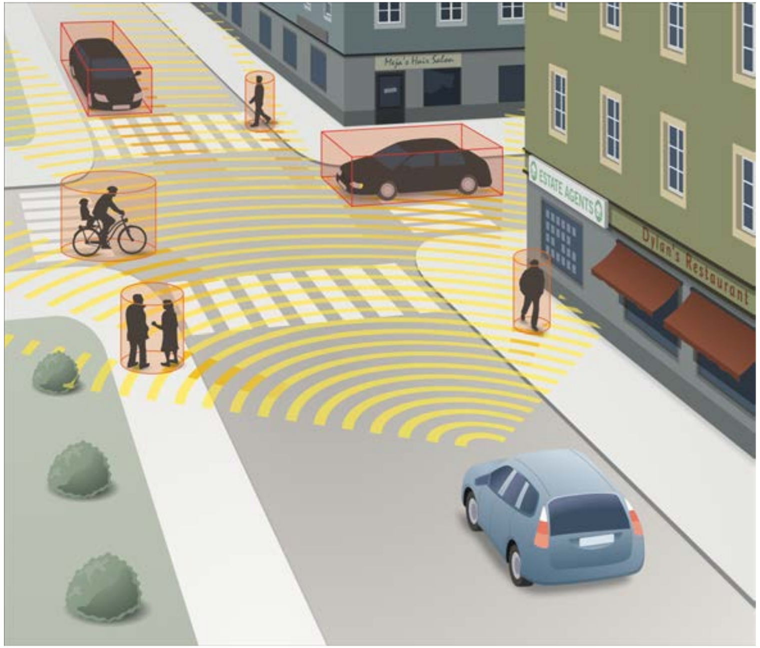


Illustration by Per Thorneus, reproduced from Granström et al (2013), "*Random set methods: Estimation of multiple extended objects*", IEEE robotics & automation magazine 21 (2), 73-82

1.2.0 Multiple Object Tracking Definitions

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Karl Granström
Department of Electrical Engineering

Definition of single object tracking

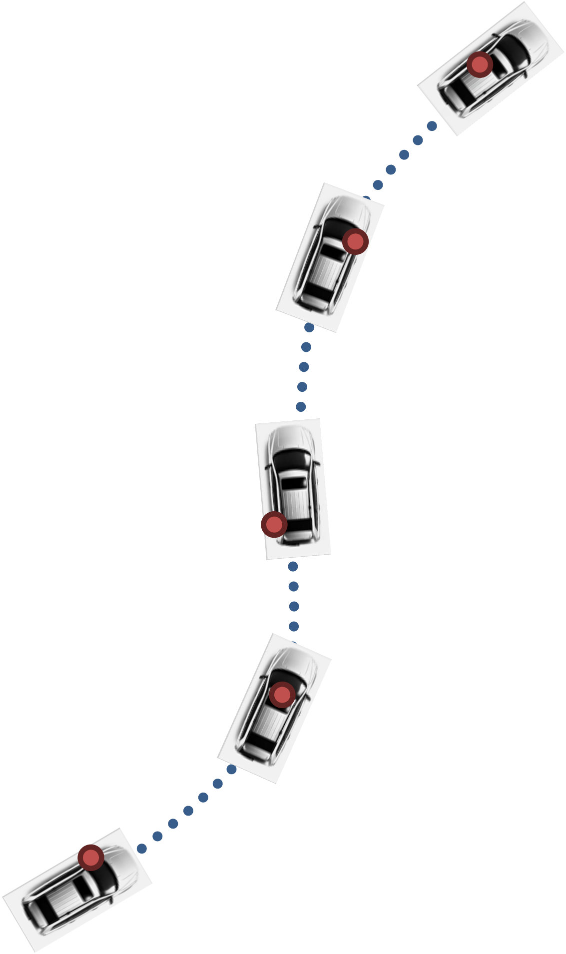
Single Object Tracking is a filtering problem.

The sequential processing of noisy sensor measurements to determine the object's state:

- Position
- Properties that describe its motion
- Other characteristics of interest

Object state is typically neither fully, nor directly, observed. **For example:**

- We want position and velocity…
- …and measure range and bearing.



Defintion of Multiple Object Tracking

Multiple Object Tracking (MOT) is the sequential processing of noisy sensor measurements to determine

- the number of dynamic objects,

and

- each dynamic object's state.

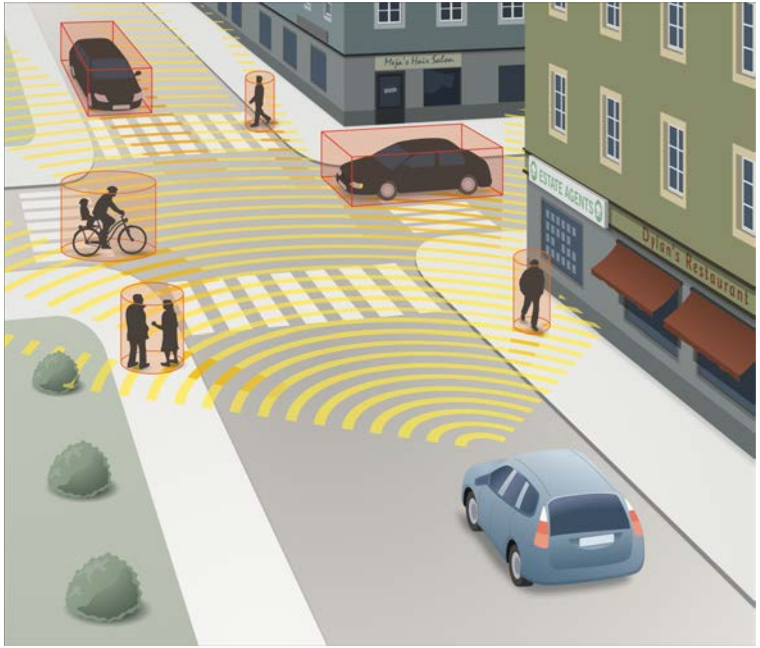
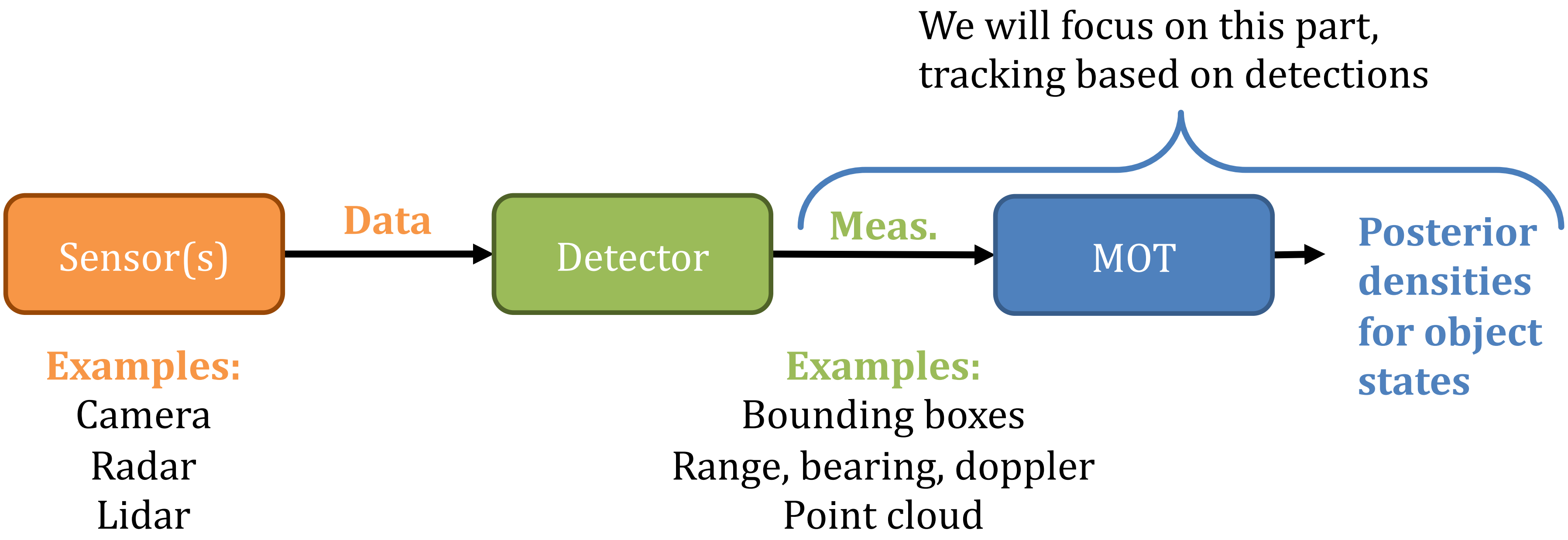


Illustration by Per Thorneus, reproduced from Granström et al (2013), "*Random set methods: Estimation of multiple extended objects*", IEEE robotics & automation magazine 21 (2), 73-82

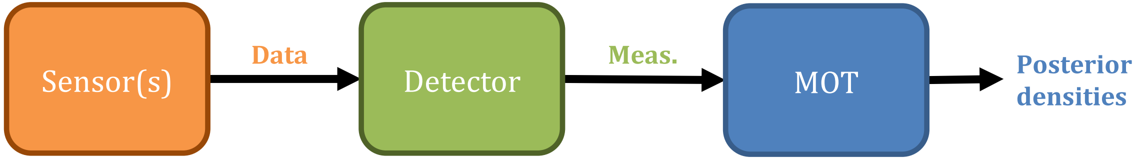
1.2.1 Tracking based on detections

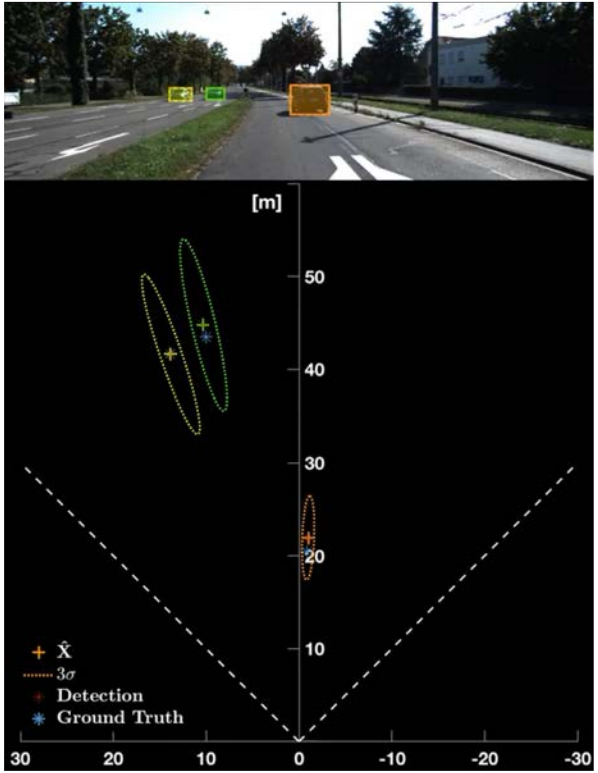
Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Multiple Object Tracking is typically based on sensor detections



Example: Vehicle tracking using mono-camera data, revisited

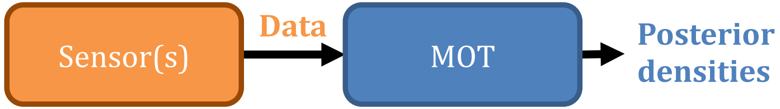




- **Sensor:** Mono-camera
- **Data:** Images
- **Detector:** constructed using Deep Learning
- **Measurements:** bounding boxes in images, and distance estimate
- **Posterior densities:** position and velocity in 3D

Track-before-detect (TBD): tracking without a detector

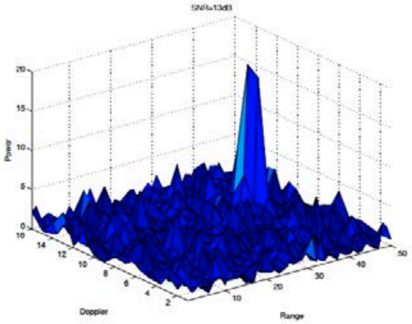
- Input raw sensor data into MOT, i.e., do not apply a detector.



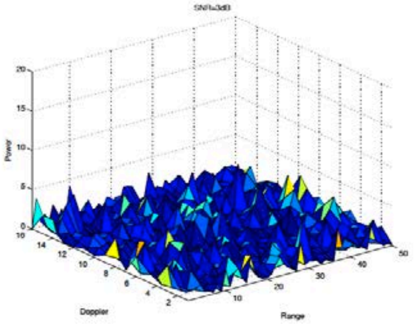
- Less common approach to tracking, and outside scope of this course.

Example:

Radar-based tracking,
low Signal-to-Noise-Ratio (SNR)



High SNR



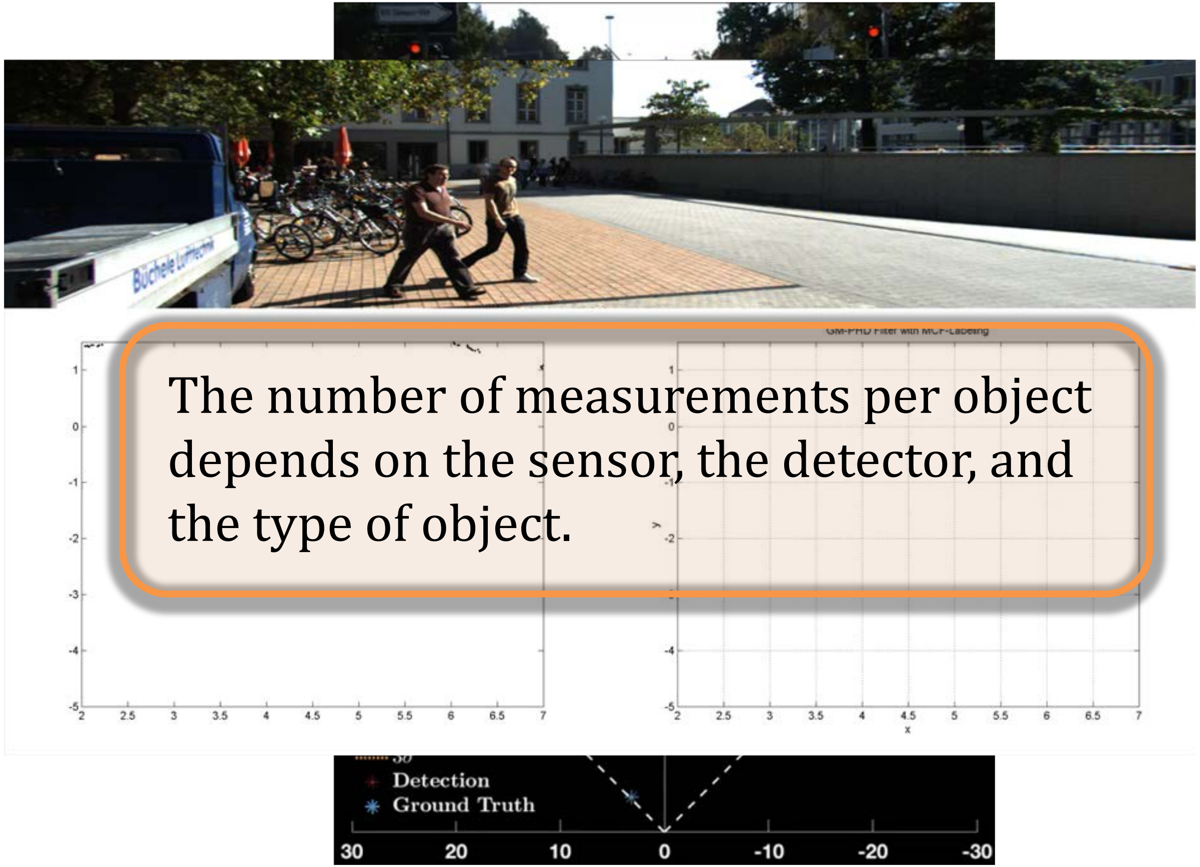
Low SNR

Torstensson & Trieb (2005), "*Particle Filtering for Track Before Detect Applications*", Master's Thesis, Department of Electrical Engineering, Linköping University

Tracking with detections: the number of detections per object

Example: Vehicles in images: at most single detection per car

Example: Pedestrians in lidar: multiple points per person

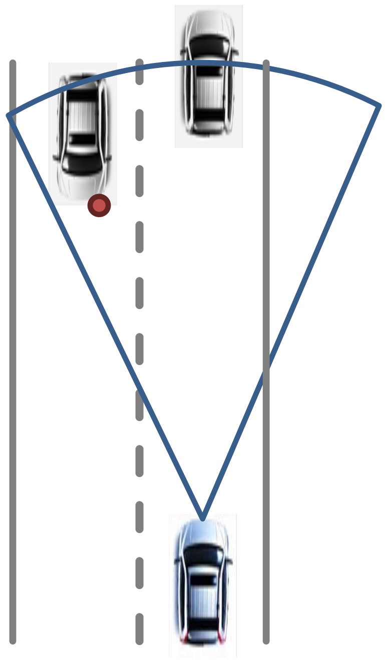


The number of measurements per object depends on the sensor, the detector, and the type of object.

1.2.2 Different types of tracking

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Point object tracking

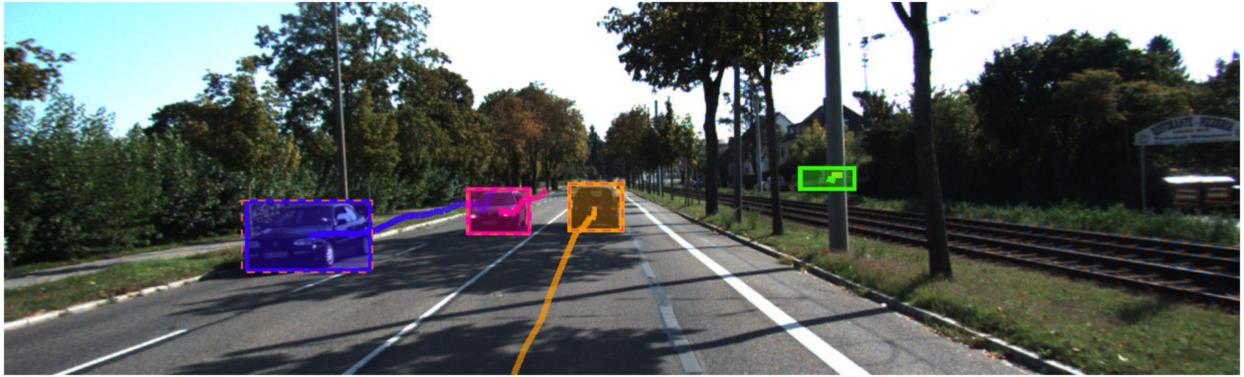


At most one detection per object, per time step

The Point Object Assumption

Examples:

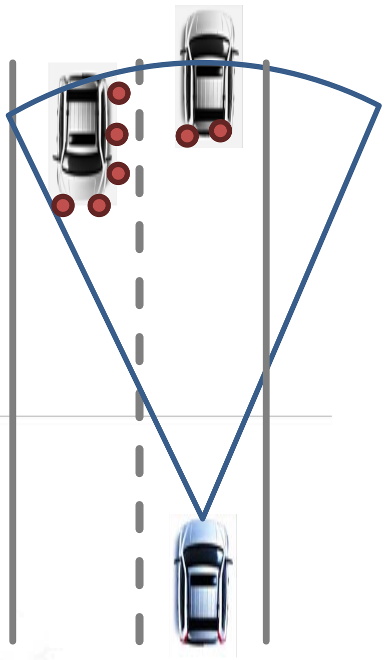
- Image detections
- Radar used for aerospace surveillance



Extended object tracking

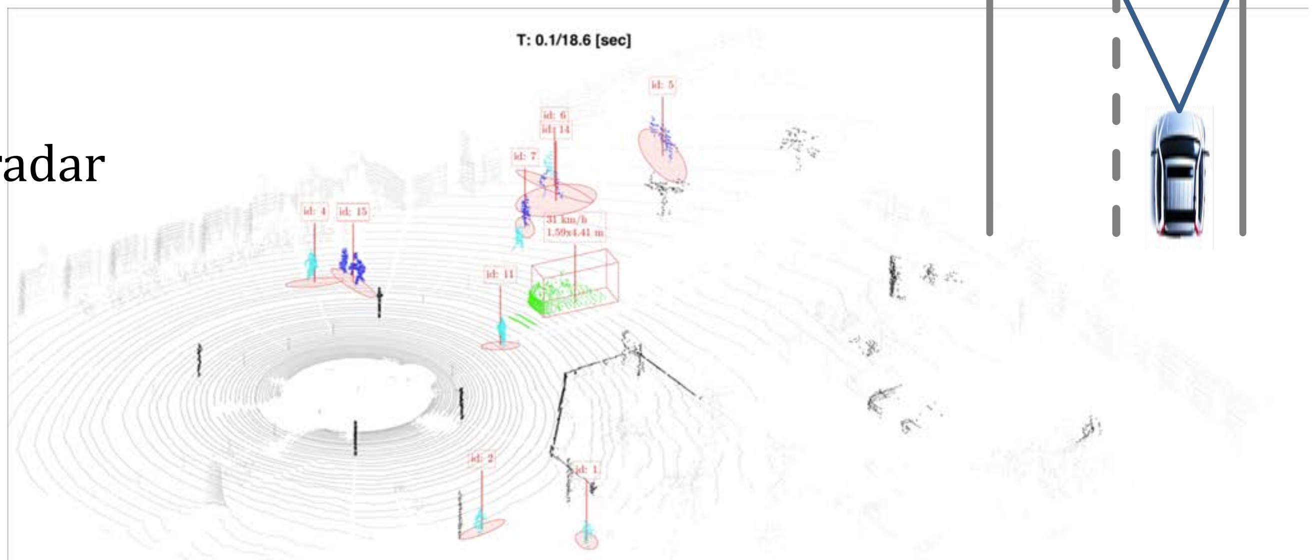
Possibly more than one detection per object, per time step.

Possible to also estimate the extent (shape and size)



Examples:

- Lidar
- Automotive radar

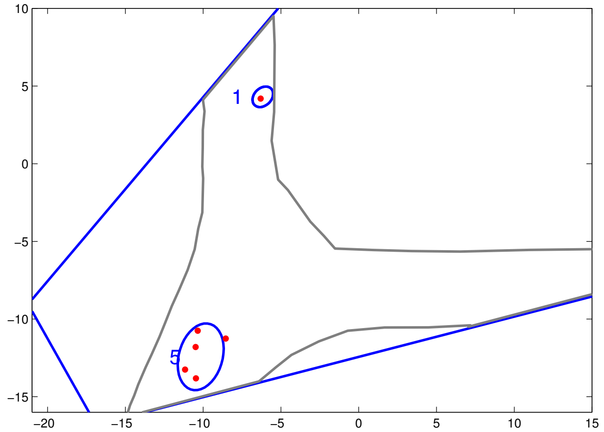


Group object tracking

Several objects treated as single entity: **a group**
Possibly more than one detection per group, per time step

Group tracking can be applied to many different sensor types.

Group tracking methods applied, e.g., when group behaviour is to be studied, or it is not feasible to track individual objects.



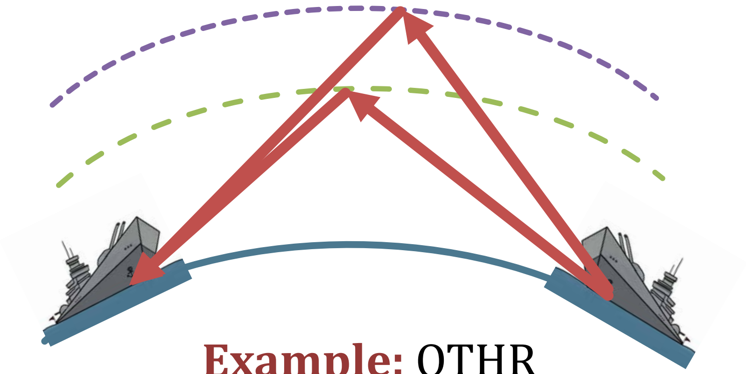
Example: pedestrian groups in video

Tracking with multipath propagation

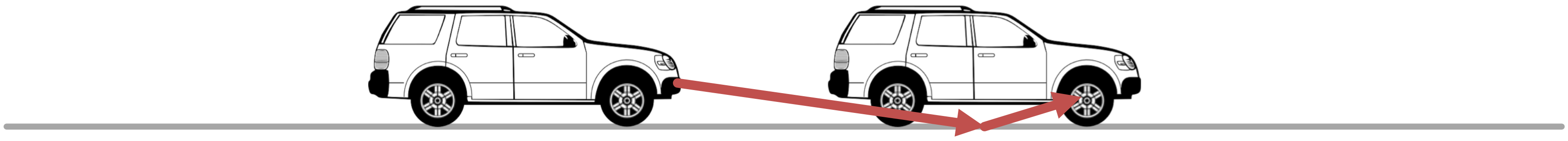
Possibly more than one detection per object, per time step. Caused by multipath phenomenon.

Examples:

- Over-The-Horizon-Radar (OTHR)
- Automotive radar



Example: OTHR



Example: Automotive radar

Tracking with unresolved targets

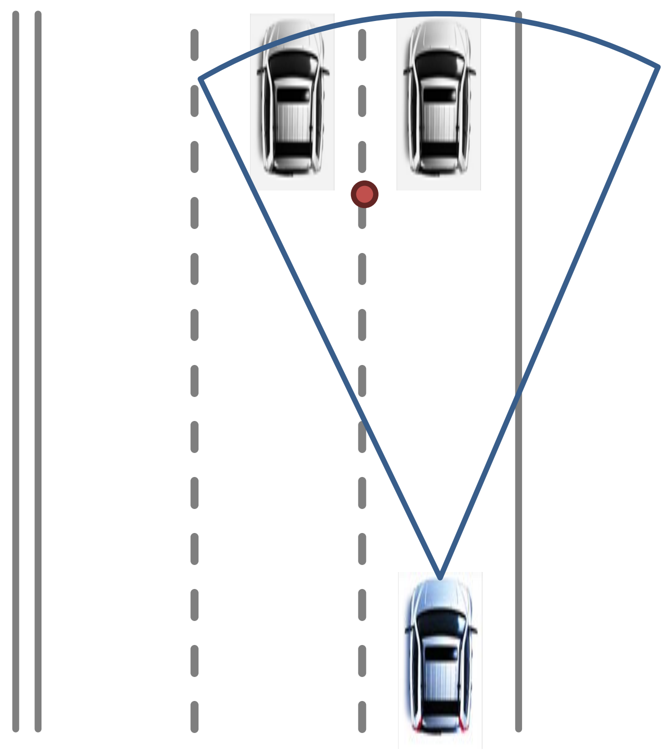
Multiple objects cause a single detection

Example:

- Automotive radar, e.g., two vehicles travelling at (approx.) the same speed.

Also called:

- Tracking with Merged Measurements.



Example: Two vehicles next to each other, travelling at the same speed.

Different types of target tracking

Focus of this course

Point object tracking
At most a single measurement per object, per time step.

Extended object tracking

Group object tracking

Tracking with multi-path propagation

Tracking with unresolved measurements

Outside scope of this course

The first three involve multiple measurements per object, but under different modelling requirements.

The last involves multiple objects per measurement.

1.3.0 Challenges in MOT

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Karl Granström
Department of Electrical Engineering

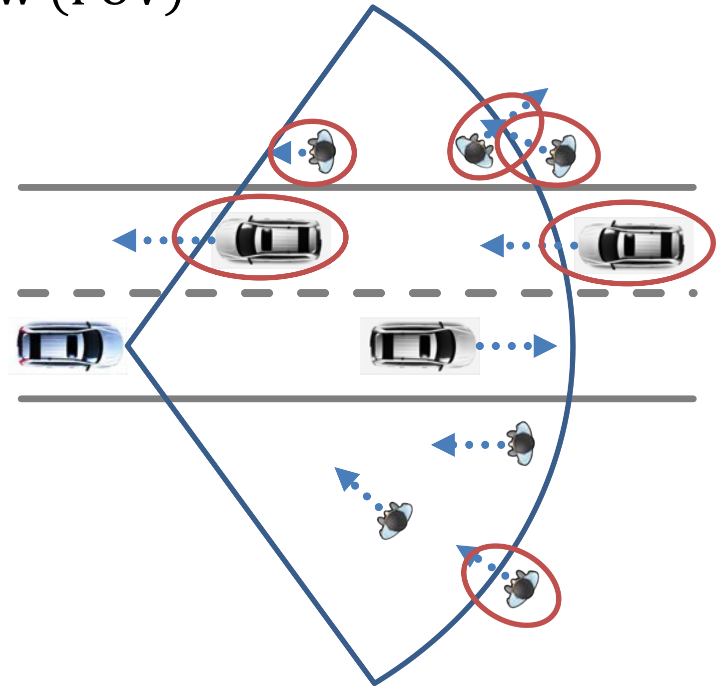
What are the challenges in MOT?

- Unknown number of objects in Field of View (FOV)
- Objects' states are unknown
- The objects move around

- Objects disappear (leave FOV)
- New objects appear (enter FOV)

- Objects inside FOV occlude one another

- The detectors are imperfect



1.3.1 Imperfect detectors

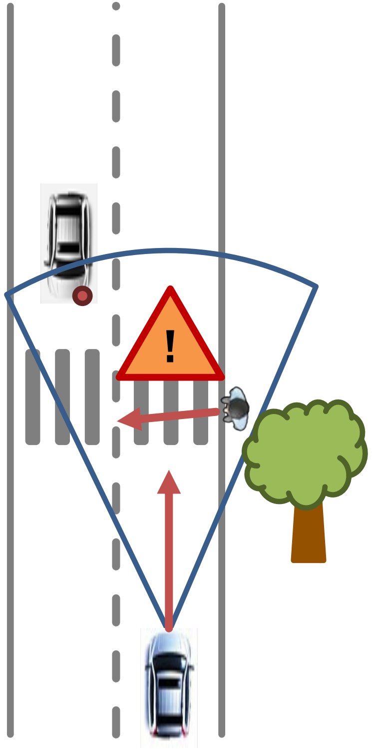
Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Detectors are imperfect: missed detections

- Missed detection: when an object is not detected by the sensor.

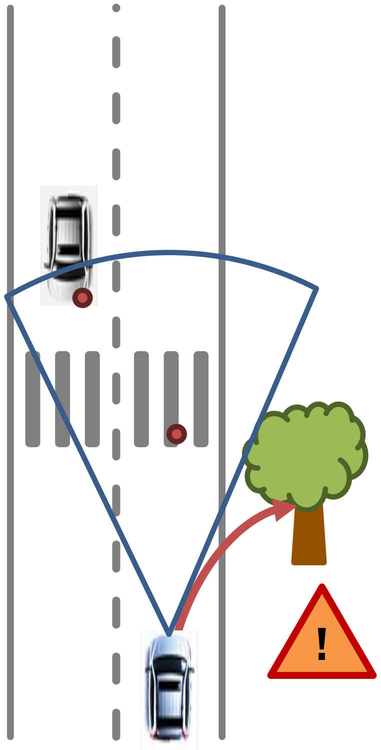
- Can, e.g., be due to
  - Environment conditions e.g., fog, poor lighting
  - Object properties
  - Occlusions

- Worst case, a missed detection leads to fatalities.



Detectors are imperfect: false detections

- False detection: a detection that is not caused by an object. Also called false alarms, or clutter.
- Can, e.g., be due to
  - Background reflects enough radar energy
  - Background looks like an object
  - Environment conditions
- Worst case, a false detection leads to fatalities.



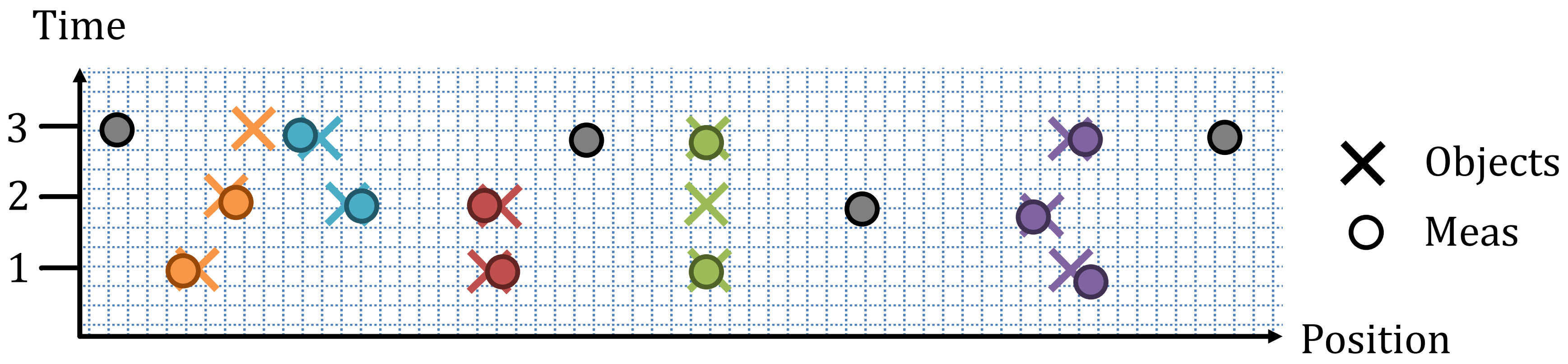
1.3.2 Data Association

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

The Data Association (DA) Problem

- No prior information about
  - which detections are from objects that we had detected before,
  - which detections are from objects that just appeared, and
  - which detections are false detections.
- Also called the Correspondence Problem
- Have to handle this uncertainty, which can be difficult:
  - Noisy sensors
  - Low probability of detection
  - Objects close to each other

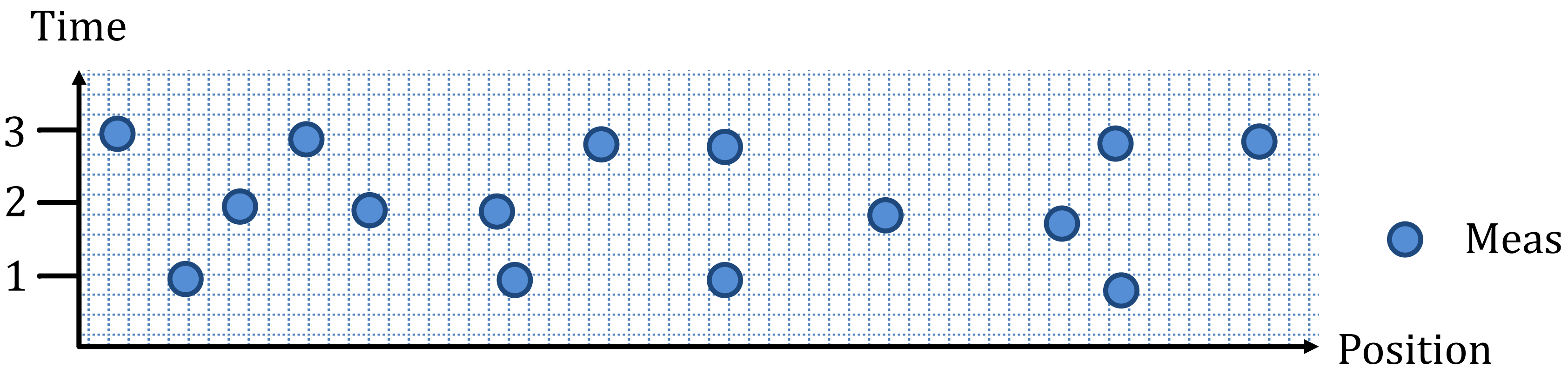
Example: simple 1D scenario, known data association



Know which measurements are from the same object, and known which measurements are clutter

Tracking multiple objects becomes easy! For each objecty, just process the corresponding measurements

Example: simple 1D scenario, unknown data association



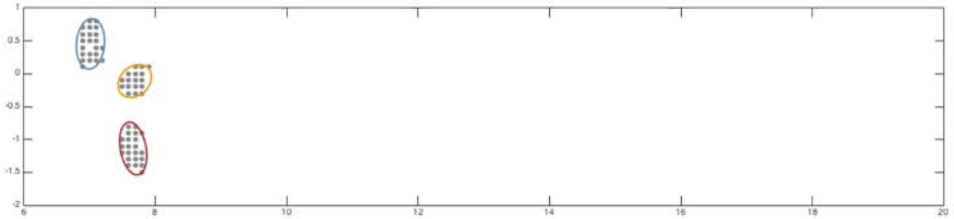
Any measurement at any time step could be a false detection, or a newly appeared object, or from an object that was detected in one of the previous time steps.

Tracking multiple objects becomes hard, because there are so many data association possibilities!

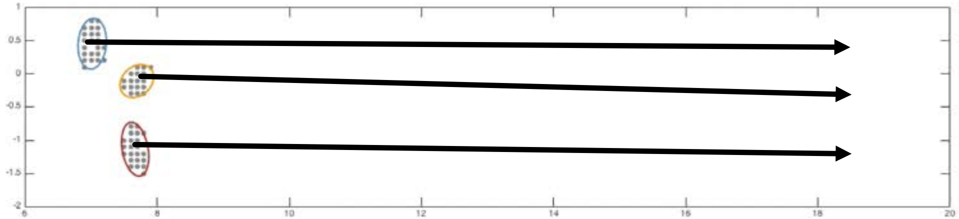
Example: Data Association (DA), data from a Lidar sensor

- Three different DA methods
- Ground truth: three persons, walking left to right
- Poor handling of the DA leads to worse tracking results

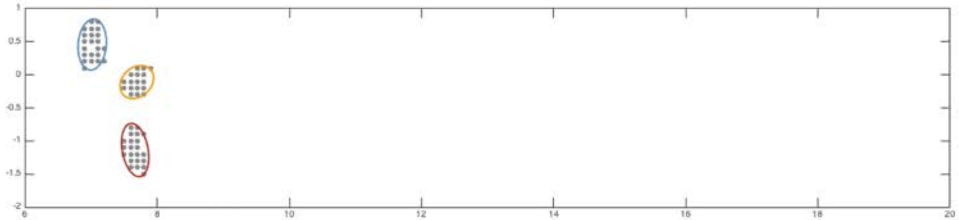
DA method 1: DBSCAN + MURTY



DA method 2: DBSCAN + k-MEANS + MURTY



DA method 3: STOCHASTIC OPTIMISATION



Granström et al (2017), " *Pedestrian tracking using Velodyne data—Stochastic optimization for extended object tracking*", Proceedings of IEEE Intelligent Vehicles Symposium
Granström et al (2018), "*Likelihood-based data association for extended object tracking using sampling methods*", IEEE Transactions on Intelligent Vehicles 3 (1), 30-45

Summary of challenges in MOT

1. Unknown, and time-varying, number of objects
2. The state of each object is unknown, and changes over time
3. The object states cannot be observed directly,
   have to be inferred from partial measurements.
4. The measurements are corrupted by noise,
   and are susceptible to missed detections and false detections.
5. Unknown data association.

1.4.0 Bayesian filtering review

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Karl Granström
Department of Electrical Engineering

Review of filtering

- MOT is generally solved using Bayesian filtering
- You are expected to be familiar with
  - Recursive estimation, Bayesian statistics, Bayes theorem
  - Linear and non-linear filtering
    - Kalman filter
    - Non-linear Kalman filters
    - Particle filters
  - Simple motion and measurement modeling
- Pre-requisites:
  - Sensor Fusion and Non-linear Filtering for Automotive Systems

1.4.1 Bayesian recursive filtering

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Bayesian recursive filtering

Two steps:

- **Prediction**: use a motion model to predict what happens from one time step to the next.
- **Update**: use measurements, and a measurement model, to update the object state densities.

The two steps are iterated: predict, update, predict, update, …

Notation

Time step: k

Object state: x_k

Measurement: z_k

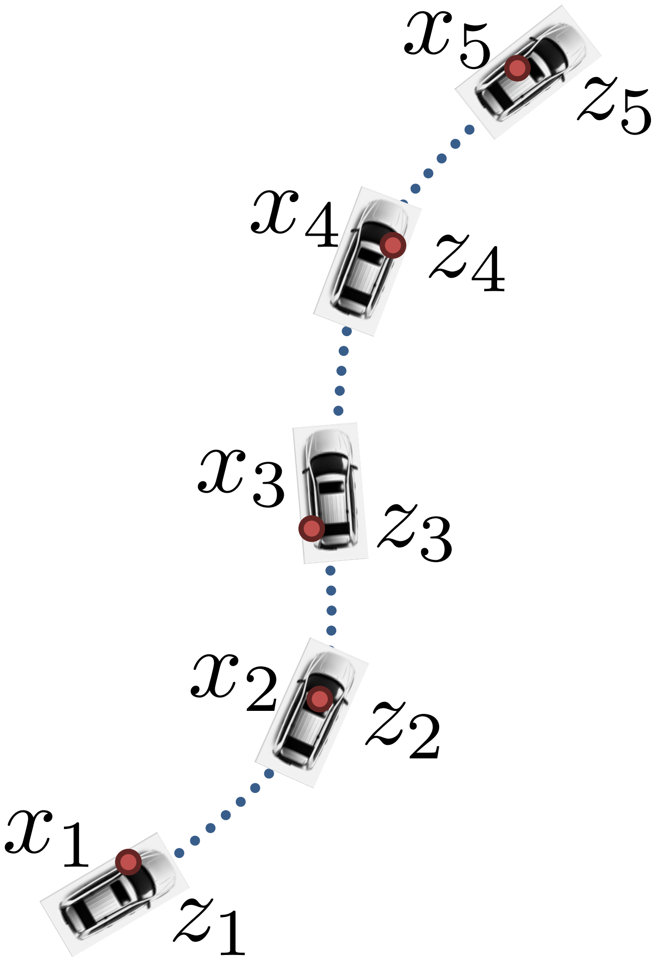
Sequence of measurements: $z_{1:\tau} = \{z_1, z_2, \ldots, z_\tau\}$

Posterior density: $p(x_k | z_{1:\tau})$

Estimate: $\hat{x}_{k|\tau}$

Gaussian pdf:

Example: Bayes filtering



1.4.2 Modelling

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Bayesian filtering needs motion and measurement modelling

- For Bayesian MOT, we need models for the multiple objects, how they move, how they enter/leave the surveillance area, and how the detector works.

- Such multiple object motion models, and multiple object measurement models, in part build upon models for single objects:
  - Single object motion model
  - Single object measurement model

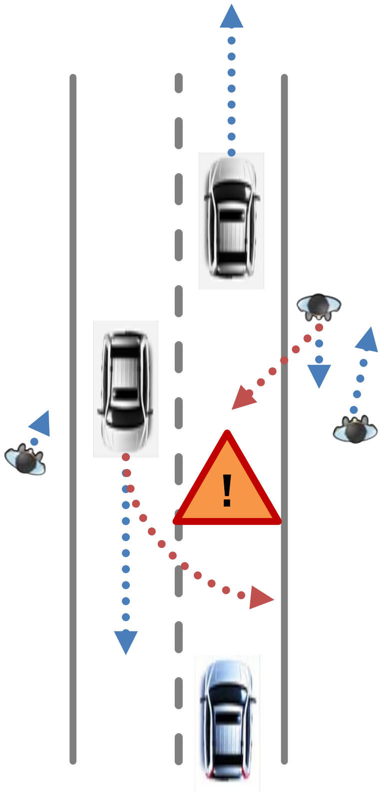
- Single object models integrated into multi-object models

1.4.3 Motion Modelling

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Single object motion modelling

- Describes how the tracked objects move; how the object state evolves over time $x_k \rightarrow x_{k+1}$
- Important to predict object motion accurately!
- Motion modelling is important also for the data association.
- Common models:
  - Linear
    - Constant velocity
    - Constant acceleration
  - Non-linear
    - Coordinated turn



Single object motion modelling

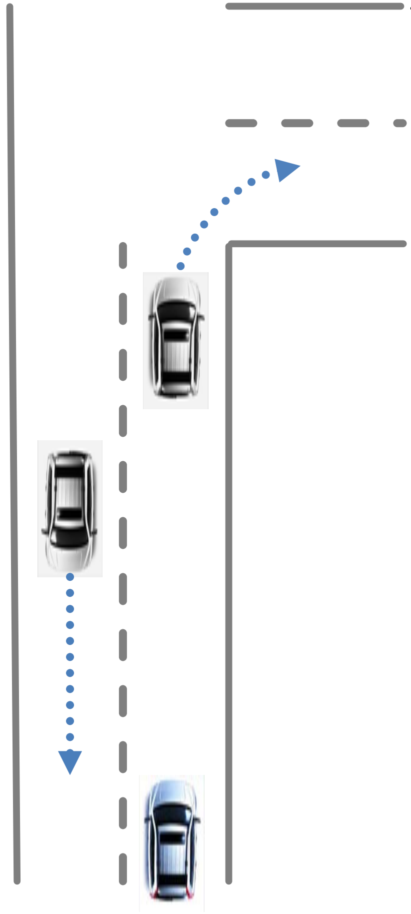
- Different types of motion, and/or different types of objects, may require different models

- Motion model with additive noise,

$$x_k = f_{k-1}(x_{k-1}) + e_{k-1}$$

The motion function can be linear or non-linear.

- Transition density $p\left(x_k | x_{k-1}\right)$



Example: Constant velocity motion model

- Consider a simple example, with a state vector two states: position and velocity,

$$x_k = \begin{bmatrix} p_k \\ v_k \end{bmatrix}$$

- The motion function, with sampling time T, is

$$x_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_{k-1} + e_{k-1}$$

Example: Gaussian distributed process noise

Assume that the process noise is Gaussian,

$$p(e_{k-1}) = \mathcal{N}(e_{k-1}\ ;\ 0,\ Q_{k-1})$$

With a motion model with additive noise,

$$x_k = f_{k-1}(x_{k-1}) + e_{k-1}$$

we get a Gaussian transition density,

$$p(x_k|x_{k-1}) = \mathcal{N}(x_k\ ;\ f_{k-1}(x_{k-1}),\ Q_{k-1})$$

1.4.4 Measurement Modelling

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Single object measurement modelling

- Describes relation between object state and measurement

$$x_k \to z_k$$

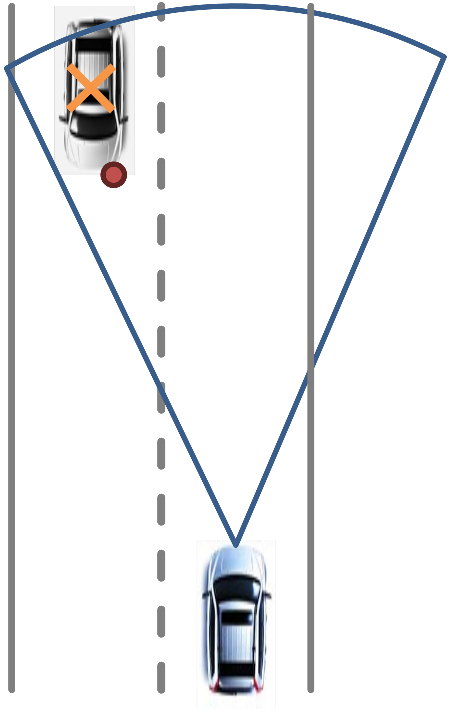
- Generally, different sensors require different modelling. Different object types may also require different models, for the same sensor.

- Measurement model with additive noise,

$$z_k = h_k(x_k) + r_k$$

The measurement function can be linear or non-linear.

- Measurement likelihood $p\left(z_k | x_k\right)$



Example: range-bearing measurements

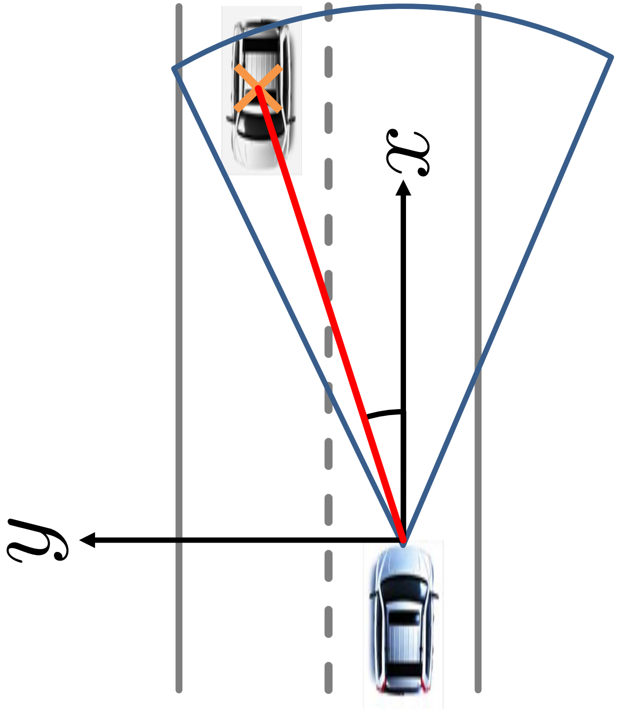
- Assume that the sensor measures the range and the bearing to the object.
- Measurement function,

$$h_k(x_k) = \begin{bmatrix} \sqrt{(p_k^x - s_k^x)^2 + (p_k^y - s_k^y)^2} \\ \tan^{-1}\left(\frac{p_k^y - s_k^y}{p_k^x - s_k^x}\right) \end{bmatrix}$$

where

$$\begin{bmatrix} p_k^x & p_k^y \end{bmatrix}^T \text{ and } \begin{bmatrix} s_k^x & s_k^y \end{bmatrix}^T$$

are the object position, and sensor position, respectively.



Example: Gaussian distributed measurement noise

Assume that the measurement noise is Gaussian,

$$p(r_k) = \mathcal{N}(r_k\ ;\ 0,\ R_k)$$

With a measurement model with additive noise,

$$z_k = h_k(x_k) + r_k$$

we get a Gaussian measurement likelihood,

$$p(z_k|x_k) = \mathcal{N}(z_k\ ;\ h_k(x_k),\ R_k)$$

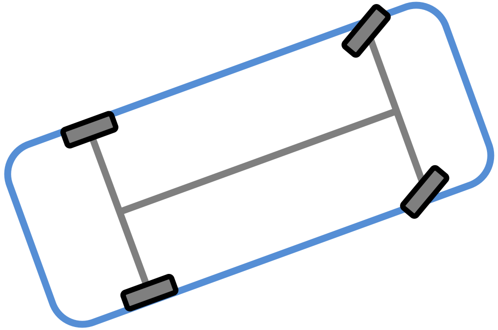
1.4.5 Sufficient Modelling

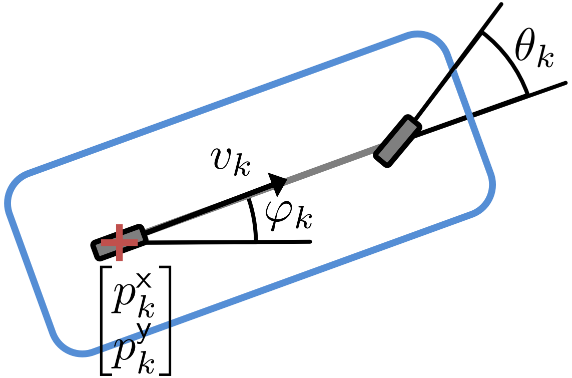
Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Sufficient modelling

- The motion and measurement models do not need to be perfect.
- Should be sufficiently accurate, and reasonably complex
  - We want high quality estimates
  - At a reasonable computational cost
- Depends on the requirements on the MOT system.
- Not the same in all scenarios.

Example: bicycle motion models are sufficient for cars





Only five states in the vector
$$x_k = \begin{bmatrix} p_k^{\mathsf{x}} \\ p_k^{\mathsf{y}} \\ v_k \\ \varphi_k \\ \theta_k \end{bmatrix} \begin{array}{l} \text{x-position} \\ \text{y-position} \\ \text{Speed} \\ \text{Heading} \\ \text{Steering angle} \end{array}$$

Simple, but versatile representation

1.4.6 Prediction, update, likelihood

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

The Bayesian filtering recursion: prediction, update and likelihood

- Chapman-Kolmogorov prediction

$$p(x_k|z_{1:k-1}) = \int p\left(x_k|x_{k-1}\right) p(x_{k-1}|z_{1:k-1})\mathrm{d}x_{k-1}$$

$k \to k-1$

- Bayes update

$$p(x_k|z_{1:k}) = \frac{p\left(z_k|x_k\right) p(x_k|z_{1:k-1})}{p(z_k|z_{1:k-1})}$$

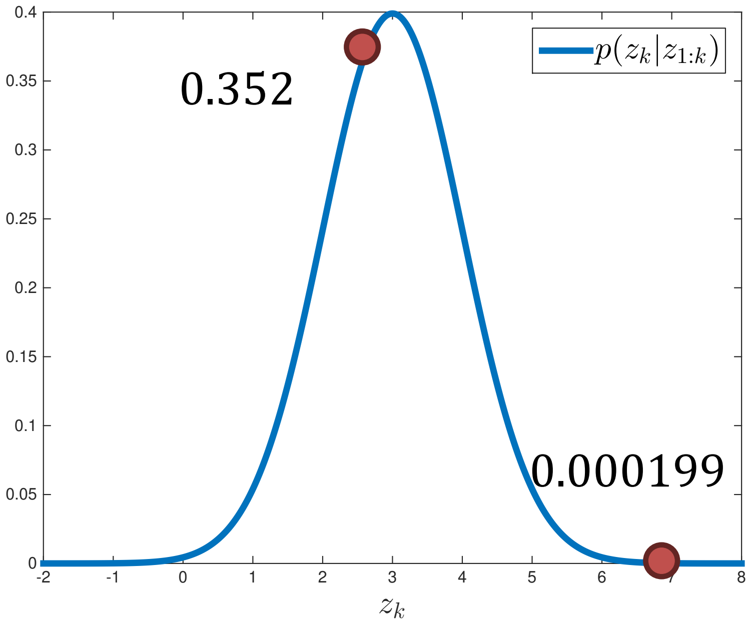
- Predicted likelihood

$$p(z_k|z_{1:k-1}) = \int p\left(z_k|x_k\right) p(x_k|z_{1:k-1})\mathrm{d}x_k$$

Predicted likelihood

$$p(z_k|z_{1:k-1}) = \int p\left(z_k|x_k\right) p(x_k|z_{1:k-1})\mathrm{d}x_k$$

Given a predicted density, and a measurement likelihood, the predicted likelihood can be used to reason about Data Association likelihoods.



Example likelihood

Example: two measurements:

$z_k^1 = 2.5$ Most likely of these two

$z_k^2 = 6.9$

1.4.7 Estimators and performance evaluation

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Object estimators

- Sometimes it is necessary to compute an object estimate using the posterior density

- For example for

  - Visualisasation of the multiple object tracking results

  - Performance evaluation, i.e., comparison to the ground truth.

Object estimators: mean and MAP

Given a posterior density $p(x_k|z_{1:\tau})$

we can extract an estimate, $\hat{x}_{k|\tau}$

Two common estimators in MOT

- the expected value (mean) estimate

$$\hat{x}_{k|\tau}^{\text{mean}} = \bar{x}_{k|\tau} = \mathbb{E}[x_k|z_{1:\tau}] = \int x_k p(x_k|z_{1:\tau})\mathrm{d}x_k$$

- the Maximum A Posteriori (MAP) estimate

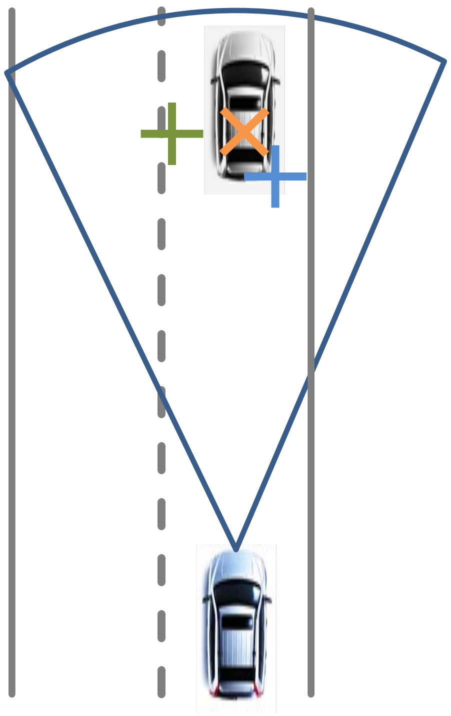
$$\hat{x}_{k|\tau}^{\text{MAP}} = \max_{x_k} p(x_k|z_{1:\tau})$$

How good is the estimate?

It is important to assess the quality of the estimate, or, in other words, to measure the performance of the MOT algorithm.

Two different estimates, marked by +/+

Which is better?



Performance evaluation

- It is important to evaluate the performance of an estimate
- Common performance measure: Mean Squared Error (MSE)

$$\mathrm{MSE}(\hat{x}_{k|\tau}) = \mathrm{E}\{(\hat{x}_{k|\tau} - x_k)^T(\hat{x}_{k|\tau} - x_k)\} = \mathrm{tr}\left\{\mathrm{E}\{(\hat{x}_{k|\tau} - x_k)(\hat{x}_{k|\tau} - x_k)^T\}\right\}$$

- Given a performance measure, we can find the estimate that gives minimal error. The Minimum MSE (MMSE) estimator is

$$\hat{x}_{k|\tau}^{\mathrm{MMSE}} = \mathrm{argmin}_{\hat{x}_{k|\tau}} \mathrm{MSE}(\hat{x}_{k|\tau})$$

1.5.0 Kalman filter review

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Karl Granström

Department of Electrical Engineering

Kalman filter

Bayesian filtering,

$$p(x_k|z_{1:k-1}) = \int p\left(x_k|x_{k-1}\right)p(x_{k-1}|z_{1:k-1})\mathrm{d}x_{k-1}$$

$$p(x_k|z_{1:k}) = \frac{p\left(z_k|x_k\right)p(x_k|z_{1:k-1})}{p(z_k|z_{1:k-1})}$$

does not have a closed form solution in general.

Important exeception (special case): when the models are linear with additive Gaussian noise, and the initial prior is Gaussian, a closed form solution is given by the Kalman filter.

If all noise is Gaussian and the models are linear, the Kalman filter is the MMSE estimator.

1.5.1 Prediction, update, likelihood

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Kalman filter

Motion model: $p(x_k|x_{k-1}) = \mathcal{N}\left(x_k\ ;\ F_{k-1}x_{k-1},\ Q_{k-1}\right)$

Measurement model: $p(z_k|x_k) = \mathcal{N}\left(z_k\ ;\ H_k x_k,\ R_k\right)$

Initial density: $p(x_0) = \mathcal{N}\left(x_0\ ;\ \bar{x}_0,\ P_0\right)$

Posterior density: $p(x_k|z_{1:\tau}) = \mathcal{N}\left(x_k\ ;\ \bar{x}_{k|\tau},\ P_{k|\tau}\right)$

Expected value: $\bar{x}_{k|\tau}$

Covariance: $P_{k|\tau}$

Kalman filter: prediction, update, likelihood

Prediction:
$$\begin{cases} \bar{x}_{k|k-1} = F_{k-1}\bar{x}_{k-1|k-1} & \text{Use model to predict} \\ P_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^{T} + Q_{k-1} & \text{Increase uncertainty} \end{cases}$$

Update:
$$\begin{cases} \bar{z}_k = H_k\bar{x}_{k|k-1} & \text{Predicted measurement} \\ \varepsilon_k = z_k - \bar{z}_k & \text{Innovation} \\ S_k = H_kP_{k|k-1}H_k^T + R_k & \text{Innovation covariance} \\ K_k = P_{k|k-1}H_k^TS_k^{-1} & \text{Kalman gain} \\ \bar{x}_{k|k} = \bar{x}_{k|k-1} + K_k\varepsilon_k & \text{Weighted average} \\ P_{k|k} = P_{k|k-1} - K_kH_kP_{k|k-1} & \text{Decrease uncertainty} \end{cases}$$

Likelihood:
$$p(z_k|z_{1:k-1}) = \mathcal{N}\left(z_k\ ;\ \bar{z}_k,\ S_k\right)$$

Non-linear models

- In MOT, it is common that the motion model and the measurement model are non-linear.

- In this case, non-linear Kalman filters can be used,
  - Extended Kalman filter (EKF)
  - Sigma-point Kalman filters
    - Unscented Kalman filter (UKF)
    - Cubature Kalman filter (CKF)

- In highly non-linear scenarios, particle filters can be used.

1.5.2 Kalman filter example

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Kalman filter example (1/2)

- Consider a simple example, with two states: position and velocity. Constant velocity motion model, T=1,

$$x_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_{k-1} + q_{k-1}$$

$$p(q_{k-1}) = \mathcal{N}\left(q_{k-1}\ ;\ \begin{bmatrix} 0 \\ 0 \end{bmatrix},\ 0.5 \begin{bmatrix} \frac{T}{3} & \frac{T}{2} \\ \frac{T}{2} & T \end{bmatrix}\right)$$

- Measurement model

$$z_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k + r_k$$

$$p(r_k) = \mathcal{N}(r_k\ ;\ 0,\ 1)$$

- Initial density

$$p(x_0) = \mathcal{N}\left(x_0\ ;\ \begin{bmatrix} 1 \\ 2 \end{bmatrix},\ \begin{bmatrix} 0.3 & 0 \\ 0 & 1 \end{bmatrix}\right)$$

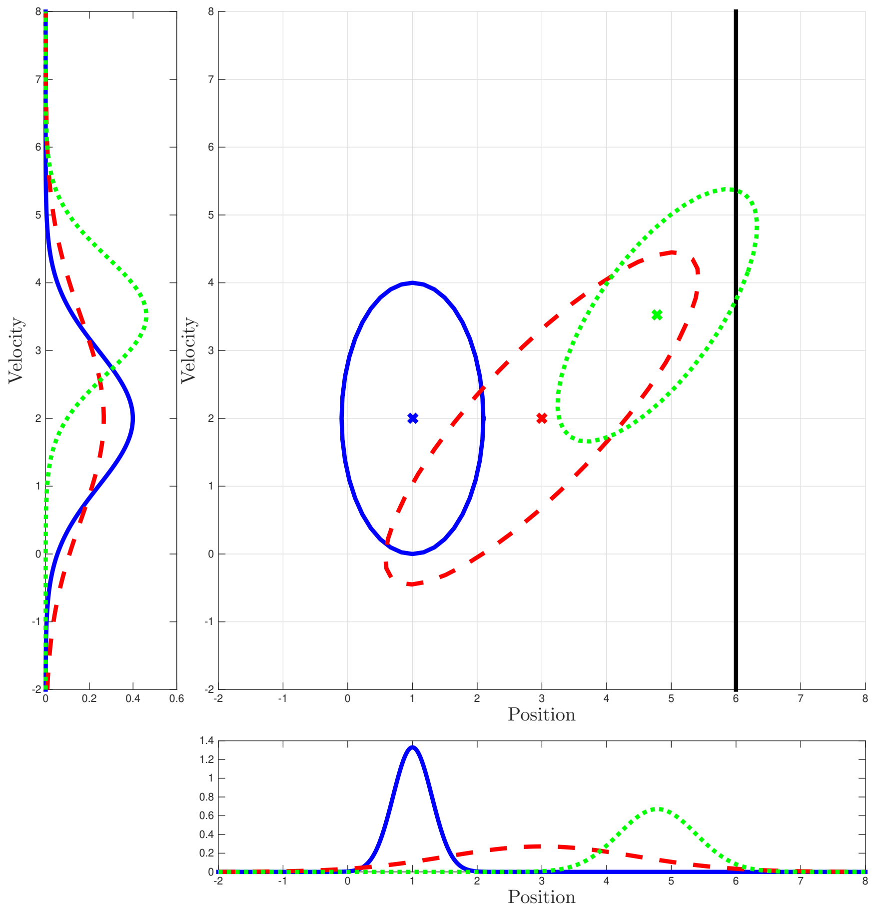
Kalman filter example (2/2)

Initial density (solid) $p(x_0)$

Prediction (dashed) $p(x_1)$

Measurement $z_1 = 6$

Update (dotted) $p(x_1 | z_1)$



1.6.0 Assumed density filtering

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Karl Granström
Department of Electrical Engineering

Consider the Kalman filter

Linear Gaussian models, Gaussian initial prior

$$p(x_k|x_{k-1}) = \mathcal{N}\left(x_k\ ;\ F_{k-1}x_{k-1},\ Q_{k-1}\right)$$
$$p(z_k|x_k) = \mathcal{N}\left(z_k\ ;\ H_k x_k,\ R_k\right)$$
$$p(x_0) = \mathcal{N}\left(x_0\ ;\ \bar{x}_0,\ P_0\right)$$

All subsequent predicted and posterior densities are Gaussian,

$$p(x_k|z_{1:k-1}) = \mathcal{N}\left(x_k\ ;\ \bar{x}_{k|k-1},\ P_{k|k-1}\right)$$
$$p(x_k|z_{1:k}) = \mathcal{N}\left(x_k\ ;\ \bar{x}_{k|k},\ P_{k|k}\right)$$

Bayesian filtering recursion gives a sequence of Gaussians

In this case, the Bayesian filtering recursion

$$p(x_0) \to p(x_1|z_1) \to p(x_2|z_1) \to p(x_2|z_{1:2}) \to \ldots \to p(x_k|z_{1:k-1}) \to p(x_k|z_{1:k})$$

results in predicting and updating the parameters of a Gaussian density.

$$\mathcal{N}_0(x_0) \to \mathcal{N}_{1|1}(x_1) \to \mathcal{N}_{2|1}(x_2) \to \mathcal{N}_{2|2}(x_2) \to \ldots \to \mathcal{N}_{k|k-1}(x_k) \to \mathcal{N}_{k|k}(x_k)$$

$$\mathcal{N}_{k|\tau}(x_k) = \mathcal{N}\left(x_k \ ; \ \bar{x}_{k|\tau}, \ P_{k|\tau}\right)$$

The mean and covariance are the sufficient statistics: sufficient to know them to know the object state density completely at any time step.

What is the significance of this?

Let the models be linear Gaussian, and let the prior be Gaussian.

The object state density is then Gaussian, and has a fixed number of parameters, of fixed dimension,

$$\bar{x}, \;\; P$$

We can track the object for as many time steps as we wish, by just updating the parameters of the Gaussian density. The number of parameters, and their dimensions, are all constant.

The complexity is predictable

1.6.1 ADF necessary for MOT

Karl Granström
Department of Electrical Engineering
Chalmers University of Technology

Is this typical in object tracking?

In many cases, either the motion model, or the measurement model, (often both!) are non-linear.

In the general case, neither prediction, nor update, is in closed form, and we do not know the exact form of the object state density.

However, we can approximate such that the object state densities are all of the same functional form – Assumed Density Filtering.

Assumed density filtering

Given an object state representation, motion and measurement models, we select a desired form for the object state density (e.g., Gaussian),

the prediction and update are approximated such that all predicted and updated object state densities are of the desired form.

Example:

$$\mathcal{N}_{k|k-1}(x_k) \approx \int \mathcal{N}\left(x_k\ ;\ f_{k-1}(x_{k-1}),\ Q_{k-1}\right) \mathcal{N}_{k-1|k-1}\left(x_{k-1}\right) \mathrm{d}x_{k-1}$$

$$\mathcal{N}_{k|k}(x_k) \propto \mathcal{N}\left(z_k\ ;\ h_k(x_k),\ R_k\right) \mathcal{N}_{k|k-1}\left(x_k\right)$$

ADF necessary for MOT

$$p(x_0) \rightarrow p(x_1|z_1) \rightarrow p(x_2|z_1) \rightarrow p(x_2|z_{1:2}) \rightarrow \ldots \rightarrow p(x_k|z_{1:k-1}) \rightarrow p(x_k|z_{1:k})$$

- For the Bayes recursion to work in practice, we need the density to be of the same form, so that in our implementation, we can apply the same prediction function and update function again.

- The predictable complexity of the tracked object's state density facilitates the implementation of the MOT system.

- ADF is a requirement for all MOT algorithms

- Everything below is just old material that is not intended to be included in the lecture.

Conjugate prior

For linear Gaussian measurement likelihoods

$$p(z|x) = \mathcal{N}\left(z\ ;\ Hx,\ R\right)$$

the Gaussian object state density,

$$p(x) = \mathcal{N}\left(x\ ;\ \bar{x},\ P\right)$$

is Conjugate Prior, meaning that the Bayes posterior

$$p(x|z) = \frac{p\left(z|x\right)p(x)}{p(z)}$$

will also be Gaussian, with mean and covariance given by the Kalman update

Definition of conjugate prior

If $\mathscr{L}$ is a class of data distributions $p(z|x)$, and $\mathscr{F}$ is a class of prior distributions for x, then the class $\mathscr{F}$ is *conjugate* for $\mathscr{L}$ if

$$p(x|z) \in \mathscr{F} \text{ for all } p(z|x) \in \mathscr{L} \text{ and } p(x) \in \mathscr{F}.$$

A list of conjugate priors can be found on Wikipedia,
https://en.wikipedia.org/wiki/Conjugate_prior

”Conjugate prediction”

Prediction is also very important in MOT.

Consider linear Gaussian transition density, and Gaussian posterior,

$$p(x|x') = \mathcal{N}\left(x\ ;\ Fx',\ Q\right)$$
$$p(x') = \mathcal{N}\left(x'\ ;\ \bar{x}',\ P'\right)$$

Then the Chapman-Kolmogorov predicted density

$$p(x) = \int p\left(x|x'\right) p(x') \mathrm{d}x'$$

is also Gaussian, with mean and covariance given by the Kalman prediction

Example of a simple bicycle model

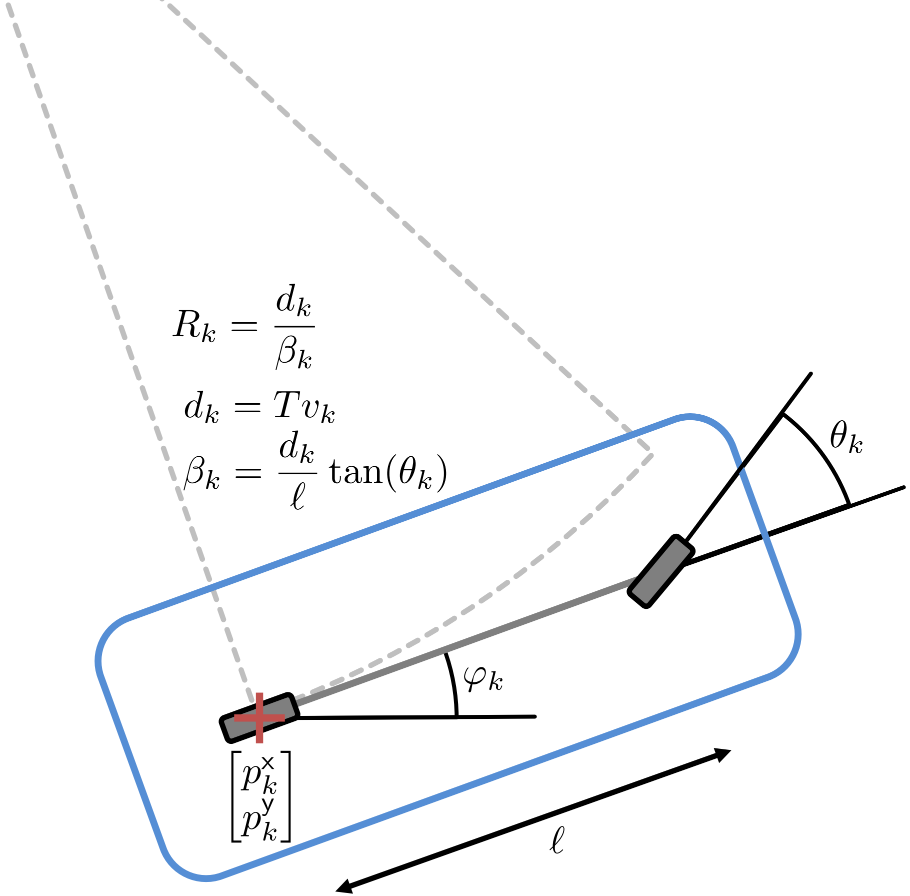
$$x_k = \begin{bmatrix} p_k^{\mathsf{x}} \\ p_k^{\mathsf{y}} \\ v_k \\ \varphi_k \\ \theta_k \end{bmatrix}$$

$$R_k = \frac{d_k}{\beta_k}$$

$$d_k = T v_k$$

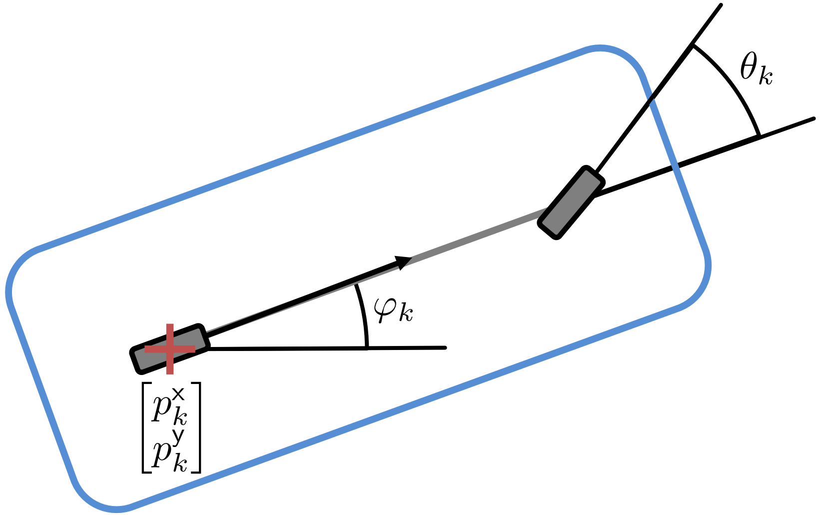
$$\beta_k = \frac{d_k}{\ell} \tan(\theta_k)$$

$$\begin{bmatrix} p_{k+1}^{\mathsf{x}} \\ p_{k+1}^{\mathsf{y}} \\ v_{k+1} \\ \varphi_{k+1} \\ \theta_{k+1} \end{bmatrix} = \begin{bmatrix} p_k^{\mathsf{x}} - R_k \sin(\varphi_k) + R_k \sin(\varphi_k + \beta_k) \\ p_k^{\mathsf{y}} + R_k \cos(\varphi_k) + R_k \cos(\varphi_k + \beta_k) \\ v_k \\ \varphi_k + \beta_k \\ \theta_k \end{bmatrix}$$



Example of a simple bicycle model

$$x_k = \begin{bmatrix} p_k^{\mathsf{x}} \\ p_k^{\mathsf{y}} \\ v_k \\ \varphi_k \\ \theta_k \end{bmatrix}$$



Bayesian filtering: prediction

Chapman-Kolmogorov prediction

$$p(x_k|z_{1:k-1}) = \int p\left(x_k|x_{k-1}\right) p(x_{k-1}|z_{1:k-1}) \mathrm{d}x_{k-1}$$

Posterior density:

Transition density:

Joint density: $p(x_k, x_{k-1}|z_{1:k-1})$

Marginalisation over the previous state:

Bayesian filtering: update

Bayes update

$$p(x_k|z_{1:k}) = \frac{p\left(z_k|x_k\right)p(x_k|z_{1:k-1})}{p(z_k|z_{1:k-1})}$$

Predicted density:

Measurement likelihood:

Bayesian filtering: likelihood

Predicted likelihood

$$p(z_k|z_{1:k-1}) = \int p\left(z_k|x_k\right) p(x_k|z_{1:k-1})\mathrm{d}x_k$$