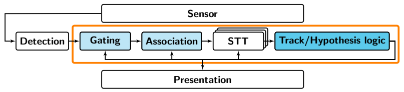
Target Tracking
Le 7: Selected topics

Gustaf Hendeby

Div. Automatic Control
Dept. Electrical Engineering
gustaf.hendeby@liu.se

Summary: lecture 5-6



Multi-Hypotheses Tracker

- The conseptual MHT given by Reid 1979.
- The Hypothesis Oriented MHT (HO-MHT).
  - Use k-best solutions to the assignment problem (Murty's method)
  - Find N_h-best hypothesis, generating as few hyps. as possible.
- Track Oriented MHT (TO-MHT).
  - Maintain tracks, create hypotheses when needed.
  - Less tracks than global hypotheses.
- Presentation of the current state is not trivial.
- MATLAB and Python frameworks for MTT.
- Guest lecture Saab: Tracking in practice.

Selected Topics

Today's lecture will focus on several different topics.

- Purpose is to highlight some problems/applications.
- The ambition is an overview with references.
- Examples: TkBD, T2T fusion, group tracking, and ETT.

However, for some topics like ETT and group tracking there might be simularities.

References on Multiple Target Tracking Topics (1/2)

- Performance Evaluation
  - A. S. Rahmathullah, Á. F. García-Fernández, and L. Svensson. Generalized optimal sub-pattern assignment metric. In *2017 20th International Conference on Information Fusion*, 2017.
  - R. Forsling, S. Julier, and G. Hendeby. Matrix-valued measures and Wishart statistics for target tracking applications. *IEEE Transactions on Aerospace and Electronic Systems*, 61(5): 12234–12244, Oct. 2025
- Track-to-Track Fusion
  - J. K. Uhlmann. Covariance consistency methods for fault-tolerant distributed data fusion. *Information Fusion*, 4(3):201–215, 2003.
  - R. Forsling, B. Noack, and G. Hendeby. A quarter century of covariance intersection: Correlations still unknown? *IEEE Control Systems Magazine*, 44(2):81–105, Apr. 2024. Lecture Notes

References on Multiple Target Tracking Topics (2/2)

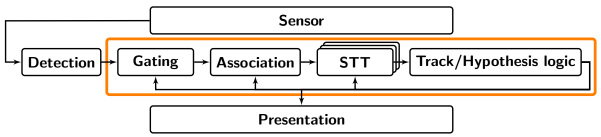
- Track Before Detect
  - B. Ristic, B.-T. Vo, B.-N. Vo, and A. Farina. A tutorial on Bernoulli filters: Theory, implementation and applications. *IEEE Transactions on Signal Processing*, 61(13):3406–3430, July 2013.
- Extended Target Tracking
  - K. Granström, L. Svensson, S. Reuter, Y. Xia, and M. Fatemi. Likelihood-based data association for extended object tracking using sampling methods. *IEEE Transactions on Intelligent Vehicles*, 3(1), Mar. 2018.
  - K. Granström, M. Baum, and S. Reuter. Extended object tracking: Introduction, overview and applications. *Journal of Advances in Information Fusion*, 12(1), Dec. 2017.
- BOOK: B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications.* Artech House, 2004.

Lecture Schedule (preliminary)

Le	Topic	Date		Ex
1	Introduction	Sept 16	15–16	
1b	Preliminaries	Sept 16	16–17	
2	Models for Target tracking	Sept 19	13–15	
3	Single target tracking	Sept 25	15–17	Ex 1
4	Multi-target tracking (1/2): GNN, JPDA	Oct 16	13–15	Ex 2
5	Multi-target tracking (2/2): MHT	Oct 30	10-12	Ex 3
6	Random Guest lecture: Per Boström-Rost (Saab)	Nov 14	13–15	
7	Various topics (TkBD, T2T, ETT)	Nov 27	15–17	
8	Random Finite Sets	**Dec 17**	**13–16**	
9	Ethical aspects	**Jan 23, 2026**	**13–15**	Seminar

- Lectures are in Large conference room in Visionen, unless otherwise stated.
- Exercises are due at the end of the course. (Doing them as the course progresses is highly recommended!)
- Dates are preliminary, check homepage and e-mail for updates.

Performance Evaluation



Single Target Tracking: root mean square error (RMSE)

- A common performance measure for estimation is the *(root) mean square error* ((R)MSE). Given M estimates $\hat{x}_{1:T}^{(i)}$ of the matching ground truth $x_{1:T}^{0(i)}$,

$$\mathsf{MSE}(\hat{x}_t) = \frac{1}{M}\sum_{i=1}^{M} \|\hat{x}_t^{(i)} - x_t^{0(i)}\|^2.$$

- The MSE combines the variance and bias of the estimate, $\mathsf{MSE}(\hat{x}_t) = \mathsf{var}(\hat{x}_t) + b_t^2$.
- **N.B.:** RMSE is a metric, MSE is not.

Single Target Tracking: RMSE performance bound

Cramér-Rao lower bound (CRLB)

The CRLB offers a fundamental performance bound for unbiased estimators and can be found as

$$\mathsf{cov}(x_t - \hat{x}_{t|t}) \succeq P_{t|t}^{\text{CRLB}},$$

where $P_{t|t}^{\text{CRLB}}$ is the CRLB, given by the EKF around the true state (parametric CRLB) and inverse intrinsic accuracy replacing all noise covariances.

It is also possible to construct a posterior CRLB.

N.B.: The CRLB can be used when setting sensor requirements and in system design.

Normalized Estimation Error Squared (NEES)

- NEES provides a consistency estimate of an estimator,

$$\mathsf{NEES}(\hat{x}_t) = \frac{1}{M}\sum_{i=1}^{M} (\hat{x}_t^{(i)} - x_t^{0(i)})^T (P_t^{(i)})^{-1} (\hat{x}_t^{(i)} - x_t^{0(i)}).$$

- Given a Gaussian assumption and correct tuning, $\mathsf{NEES}(\hat{x}_t) \sim \chi^2(n_x)$
  - $< \boldsymbol{n_x}$ conservative estimate, *i.e.*, the estimate is better than indicated with the P.
  - $\approx \boldsymbol{n_x}$ the estimated covariance matches what is observed.
  - $> \boldsymbol{n_x}$ optimistic estimate, *i.e.*, the estimate is worse than indicated with the P.

N.B.: A $\chi^2(n_x)$ distribution has mean n_x and variance $2n_x$.

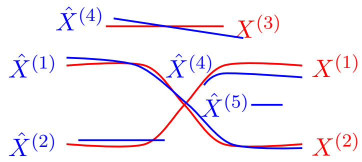
Multi-Target Tracking: performance

Multi-target tracking performance is a problem of relating elements of two different sets:

$$\{X^{(1)}, \dots, X^{(N)}\} \overset{\varphi: n \leftrightarrow m}{\longleftrightarrow} \{\hat{X}^{(1)}, \dots \hat{X}^{(M)}\}$$

How to handle:

- Inconsistent number of targets? $N \neq M$
- Match estimated track to ground truth track? φ
- Label switches? φ changes over time



How to judge the tracking result (blue tracks), compared to the ground truth (red tracks)? The number of tracks does not match, and the labels are different...

Multi-Target Tracking: performance criteria

Important properties:

- RMSE/NEES per target; how accurate are estimated tracks?
- Time to start track; how long does it take to confirm a new track?
- Track consistency; are the tracks kept together over time?

GOSPA (1/2)

- *Generalized optimal subpattern assignment* (GOSPA) metric is an extension of RMSE to the multi-target setting.
- Two sets of tracks $X = \{x^{(i)}\}_{i=1}^N$ (ground truth) and $\hat{X} = \{\hat{x}^{(i)}\}_{i=1}^M$ (estimated tracks).
- Is local, in the sense that is does not take label switches into consideration.
- Cardinality (number of targets) mismatch is penalized.
- Supersedes *optimal subpattern assignment* (OSPA), which has a slightly different carnality handling.

GOSPA (2/2)

GOSPA metric

Given two sets of tracks $\hat{X}$ and X, a metric $d(x, \hat{x})$, and a cost for incorrect targets c,

$$\tilde{d}_p^{(c,\alpha)}(X, \hat{X}) = \Big(\min_{\varphi \in \Phi_{|\hat{X}|}} \sum\nolimits_{i=1}^{|X|} d^{(c)}(x^{(i)}, \hat{x}^{(\varphi(i))})^p + \tfrac{c^p}{\alpha}(|\hat{X}| - |X|)\Big)^{\frac{1}{p}},$$

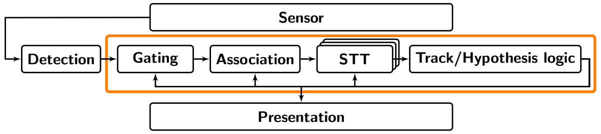
$|X| \leq |\hat{X}|$ else $\tilde{d}_p^{(c,\alpha)}(X, \hat{X}) = \tilde{d}_p^{(c,\alpha)}(\hat{X}, X)$ where $d^{(c)}(x, \hat{x}) = \min\big(d(x, \hat{x}), c\big)$ is a version of the chosen norm that saturates at c.

Usually $\alpha = 2$ and $p = 2$.

GOSPA: observatories

- Comparing different aspects are always difficult! Changing the cut-off c, can drastically change the results.
- The GOSPA error can be divided into separate parts:
  - Estimation errors (roughly the RMSE).
  - Cost of missed targets.
  - Cost of false targets.
- The pure GOSPA compares time instances independently, but can be extended to a track formulation also punishing identity switches.

Track-to-Track Fusion



Track-to-Track (T2T) Fusion

- Consider a network of stand alone nodes performing target tracking.
- Estimates are passed around, which can lead to double use of data.
- How to efficiently combine measurements in a sound way?

Independent Estimates

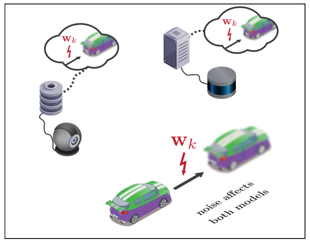
Sensor Fusion Formula

Independent estimates $\{(\hat{x}^{(i)}, P^{(i)})\}_i$ we can combine these using the fusion formula:

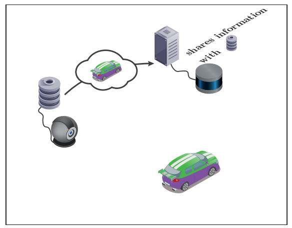
$$\hat{x} = P \sum_i (P^{(i)})^{-1} \hat{x}^{(i)}$$
$$P^{-1} = \sum_i (P^{(i)})^{-1}.$$

In case of dependent estimates, more elaborate methods are needed.

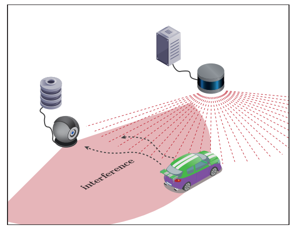
Sources of Dependence



Common process noise.



Common information.



Correlated sensor noise.

Problem

Reusing data results in both **misleading estimates** and **incorrect uncertainty** information.

Typically the estimates are indicated as **too certain**.

Dependent Measurements (1/2)

Covariance Intersection (CI)

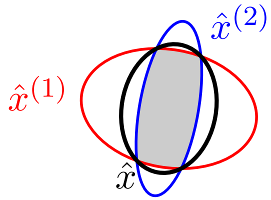
A conservative estimate of combined estimate of several estimates $\{(\hat{x}^{(i)}, P^{(i)})\}_i$ with unknown correlations:

$$\hat{x} = P \sum_i \omega^{(i)} (P^{(i)})^{-1} \hat{x}^{(i)}$$
$$P^{-1} = \sum_i \omega^{(i)} (P^{(i)})^{-1},$$

where $\sum_i \omega^{(i)} = 1$ are chosen as to minimize P under some norm, usually $\operatorname{tr}(P)$ or $\det(P)$.

Covariance Intersection: illustration

- The covariance of the fused estimate will be within the intersection between the two covariances.
- Covariance intersection will choose the "smallest" P, covering the intersection.



Dependent Measurements (2/2)

Safe Fusion

An easy to compute, but not completely conservative method to fuse two estimates with unknown dependencies.

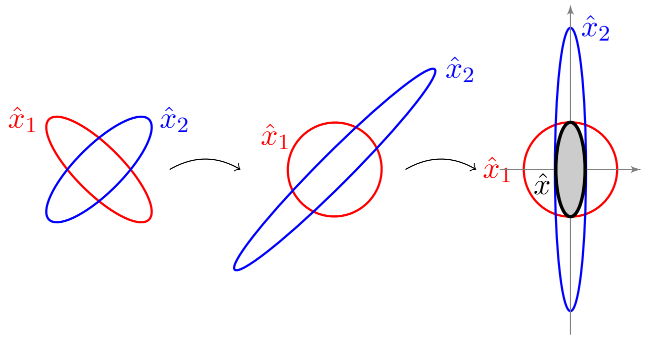
1. SVD: $P^{(1)} = U_1 D_1 U_1^T$.
2. SVD: $D_1^{-1/2} U_1^T P^{(2)} U_1 D_1^{-1/2} = U_2 D_2 U_2^T$.
3. Transformation matrix: $T = U_2^T D_1^{-1/2} U_1^T$.
4. State transformation: $\hat{\bar{x}}_1 = T\hat{x}^{(1)}$ and $\hat{\bar{x}}_2 = T\hat{x}^{(2)}$.
   The covariances of these are cov$(\hat{\bar{x}}_1) = I$ and cov$(\hat{\bar{x}}_2) = D_2$.
5. For each component $i = 1, 2, \dots, n_x$, let

$$[\hat{\bar{x}}]_i = [\hat{\bar{x}}_1]_i, \quad [D]_{ii} = 1 \quad \text{if} \quad [D_2]_{ii} \geq 1,$$
$$[\hat{\bar{x}}]_i = [\hat{\bar{x}}_2]_i, \quad [D]_{ii} = [D_2]_{ii} \text{if} \quad [D_2]_{ii} < 1.$$

6. Inverse state transformation:

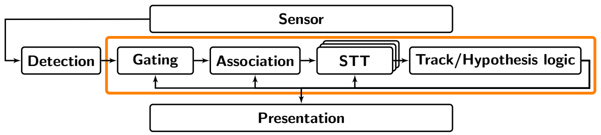
$$\hat{x} = T^{-1}\hat{\bar{x}}, \quad P = T^{-1} D^{-1} T^{-T}$$

Safe Fusion: illustration



- The two estimates are transformed to become as independent as possible.
- Extract the best information in each direction.

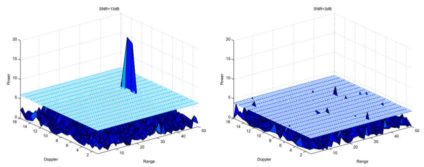
Track Before Detect (TkBD)



LINKÖPING UNIVERSITY

Track Before Detect: SNR motivation

General TkBD concept: simultaneous detection and tracking.

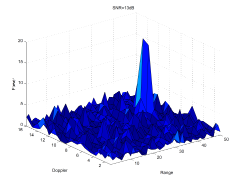


- High SNR: traditional detection works.
- Low SNR: traditional detections will not work.

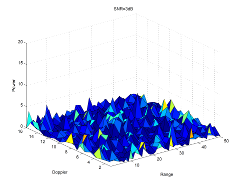
Note:

The typical detectors used are *constant false alarm rate* (CFAR). For feasible solutions, the detection threshold cannot be set too low. . .

Track Before Detect: idea



(a) SNR=13 dB. A high SNR makes it easy to detect the point target.



(b) SNR=3 dB. A low SNR makes the target hard to detect in a cluttered environment.

- Radar example (but also applies for other sensors).
- For simplicity of argument, assume one target.
- Consistent motion model.
- Applicable to low energy (stealthy) targets.

Track Before Detect: assumptions and methods

Basic assumptions:

- Collect data over several scans to enhance weak targets.
- Prohibit or penalize deviations from modeled motion.
- Assume one target (or sufficiently separated).

Ways to achieve TkBD:

- Batch-algorithms
- Hough transform
- Dynamic Programming
- **Bayesian filtering** (often Bernoulli filters)

Track Before Detect: Bayesian concept (1/2)

Consider a target moving in 2D with its intensity as part of the state

$$x_t = \begin{pmatrix} \mathsf{x}_t & \mathsf{y}_t & v_{\mathsf{x},t} & v_{\mathsf{y},t} & I_t & m_t \end{pmatrix}^T$$

Dynamic model:

- Kinnematics according to a CV-model or similar. (The model needs to be restrictive for TkBD to work well.)
- Existence, m, is modeled according to a Markov, with birth/death according to:

$$P_b = \Pr(m_t = 1 | m_{t-1} = 0)$$
$$P_d = \Pr(m_t = 0 | m_{t-1} = 1).$$

Track Before Detect: Bayesian concept (2/2)

Observation model:

$$y_t^{(i,j)} = \begin{cases} h^{(i,j)}(x_t) + e_t^{(i,j)}, & \text{if target present, } m = 1 \\ e_t^{(i,j)}, & \text{if target absent, } m = 0 \end{cases}$$

where $h^{(i,j)}(x_t)$ is the target intensity contribution in pixel (i, j).

- The "measurement" is the full measurement volume $y_t = \big(y_t^{(i,j)}\big)_{i,j}$, *i.e.*, the whole picture.
- It is often useful to model the measurement as $p(y_t | x_t)$.

Solve the resulting Bayesian estimation problem for the position and existence, *e.g.*, using a particle filter.

Track Before Detect: radar example (1/2)

Now consider a radar tracking stealthy targets:

- Instead of thresholding, the entire radar video signal is used, *i.e.*, the received power, $P(r^{(j)}, d^{(k)}, b^{(l)})$, $\forall j, k, l$ in each cell. (Range: r, Doppler: d, Bearing: b)
- The measurements consist of the power levels in $N_r \times N_d \times N_b$ sensor cells, where N_r, N_d, and N_b are the number of range, Doppler, and bearing cells.

For each range-Doppler-bearing cell, $(r^{(j)}, d^{(k)}, b^{(l)})$, the received power in the measurement relation is given by

$$y_{P,t}^{jkl} = \left| y_{A,t}^{jkl} \right|^2 = |A_t^{jkl} \cdot h_A^{jkl}(x_t) + e_t^{jkl}|^2,$$

where $j = 1, \dots, N_r,\ k = 1, \dots, N_d,\ l = 1, \dots, N_b$.

Track Before Detect: radar modeling (2/2)

$$h_A^{jkl}(x_t) = e^{-\frac{(r^{(j)} - r_t)^2}{2R}\lambda_r} e^{-\frac{(d^{(k)} - d_t)^2}{2D}\lambda_d} e^{-\frac{(b^{(l)} - b_t)^2}{2B}\lambda_b}.$$

The constants R, D, and B are related to the size of the range cell, the Doppler cell, and the bearing cell. Losses are represented by the constants λ_r, λ_d, and λ_b. The noise is defined by

$$e_t^{jkl} = e_{I,t}^{jkl} + \imath \cdot e_{Q,t}^{jkl},$$

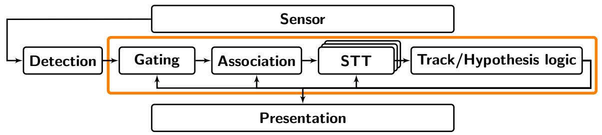
which is complex Gaussian, where $e_{I,t}^{jkl}$ and $e_{Q,t}^{jkl}$ are independent, zero-mean white Gaussian with variance σ_e^2, for the in-phase and quadrature-phase, respectively.

Solve using a particle filter, when some of the particles represent no target, and some target existing. As a target appears, the particles representing an existing target will start to dominate.

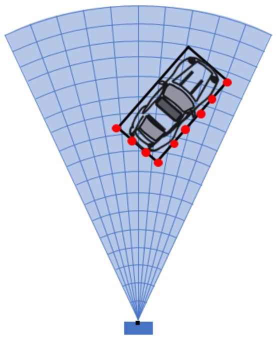
Track Before Detect: summary

- TkBD can be used for extended targets.
- Computational intensive.
- Motion model must correspond to true target.
- Multiple targets is more complicated.
- Possible to track in low SNR.

Extended target Tracking (ETT)



Extended Target Tracking



From MATLAB Sensor fusion and tracking toolbox.

When the sensor resolution becomes higher than the target size:

- Target cannot be modeled as points anymore.
- One measurement per target does not hold any more.
- Measurement could be correlated.
- Options to deal with this:
  - Cluster the measurements before applying a regular tracker.
  - Take the target extent into consideration (estimate it).

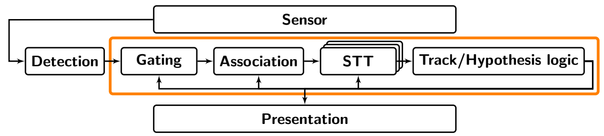
Extended Target Tracking: measurement clustering

- A standard MTT is a point target tracker.
- It assumes that every track can be detected at most once by a sensor in a scan.
- If detections are not clustered, the tracker generates multiple tracks per object.
- Clustering returns one detection per cluster, at the cost of having a larger uncertainty

Extended Target Tracking: extension modeling

- **Geometry:** Need to specify a model for the extended object: rectangular, ellipsoidal, star convex etc.
- **Dynamics:** Each extended object must have some motion model, for instance coordinated turn about its pivot.
- ETT handles multiple detections per object and sensor without the need to cluster detections, at the cost of more advanced association and a more complex model.

Group Tracking



Group Tracking

Standard tracking:

- A target is a “single point”.
- We receive at most one measurement for each target.

Group tracking:

- Tracking a group of targets that moves in a similar way.
- An extended target could be seen as a similar problem.

N.B.: extended target tracking and group tracking could sometimes be the same.

Group Tracking: dynamic model

Consider the bulk model (B) and the individual targets x, according to:

$$
\begin{aligned}
B_{t+1} &= f^B(B_t, w_t) \\
x_{t+1}^{(i)} &= f^{(i)}(x_t^{(i)}, w_t^{(i)}),
\end{aligned}
$$

where we assume $i = 1, \dots, N_{tg}$. Usually $f^{(i)} = f$.

Note: The bulk is the center or the mean position, orientation etc. Everything can be implemented by extending the state vector.

Group Tracking: observation model

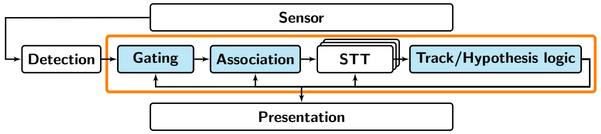
The observation cannot originate from multiple sources. Each measurement is from a target or clutter

$$y_t^{(j)} = h\big(\Psi(x_t^{(i)}, B_t)\big) + e_t,$$

where Ψ be a nonlinear transformation.

Now proceed with association etc.

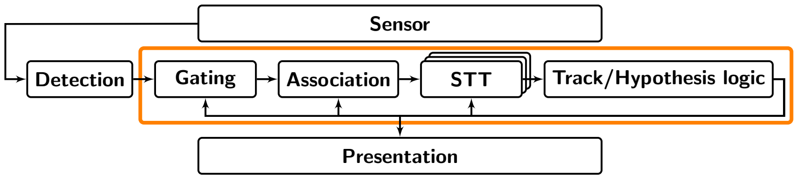
Summary Classic Target Tracking



Summary Multi-Target Tracking Course: basis

Problem formulation:

Multi-target tracking is the problem of decide how many targets are present and how they move, given measurements with imperfections.



Components in classical multi-target tracking solutions.

Summary Multi-Target Tracking Course: single target tracking

Single target tracking

- Filters
  - (Extended/Unscented) Kalman type filter
  - Particle filter
  - Filter banks (IMM, GBP, RPEKF, . . .)
- Motion models: $x_{t+1} = f(x_t) + v_t$
  - Constant velocity
  - Constant acceleration
  - Coordinated turn
  - Switched models for maneuvering targets
- Observation models: $y_t = h(x_t) + e_t$
- Clutter
- Missed detections

Summary Multi-Target Tracking Course: multi-target tacking

Multi-target tracking

- Classic methods (GNN, JPDA, MHT):
  - Differ in the association method used.
  - Track logic for initiation and termination.

Which MTT Method to Use?

Computation \ SNR	Low	Medium	High
Low	Group TT	GNN	GNN
Medium	MHT	GNN / JPDA	GNN
High	TrBD / MHT	MHT	Any

- GNN and JPDA are very bad in low SNR.
- When using GNN, one generally has to enlarge the overconfident covariances to account for neglected data association uncertainty.
- JPDA has track coalescence and should not be used with closely spaced targets, see the "coalescence avoiding" versions.
- MHT requires significantly higher computational load but it is said to be able to work reasonably under 10–100 times worse SNR.

Summary Multi-Target Tracking Course: extensions

- Track Before Detect: raw observations are used for simulataneous detection and tracking in **poor SNR**.
- Performance measures
  - Root mean square error (RMSE)
  - Normalized estimation error square (NEES)
  - Cramér-Rao lower bound (CRLB)
  - Generalized optimal subpattern association (GOSPA): multi-target
- Extended target and group tracking
- Various examples of tracking applications from research and industry