FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Assignment of master's thesis

Title:	Estimation of detection probability in multitarget filters using object advanced image processing techniques
Student:	Bc. Michal Seibert
Supervisor:	doc. Ing. Kamil Dedecius, Ph.D.
Study program:	Informatics
Branch / specialization:	Knowledge Engineering
Department:	Department of Applied Mathematics
Validity:	until the end of summer semester 2024/2025

Instructions

Abstract: The subject of the thesis is the dynamic adjustment of the detection probability in random finite set (RFS)-based filters. These filters can track objects in noisy environments from imprecise measurements. However, they rely on the knowledge of the object (mis)detection probability. If this quantity is set inappropriately, the filters are oversensitive and prone to track loss. However, there is no convenient methodology for a consistent estimation of the detection probability. The present thesis aims to focus on its inference using algorithms for object detection and image segmentation. For object detection, the YOLO model should be used, for segmentation, the Segment anything from Meta AI is a possible way towards a solution. These models can recognize various objects in the image. It is conjectured that the combination of these algorithms can yield a filter with good robustness to target misdetections.

The goals are as follows:
- study the principles of the multitarget tracking algorithms
- study the principles of image segmentation and object detection
- propose a technique for estimation of object detection probability
- perform assessment of the proposed algorithm and discuss the obtained results

Literature:
[1] A. F. Garcia-Fernandez, A. S. Rahmathullah, and L. Svensson, "A Metric on the Space of Finite Sets of Trajectories for Evaluation of Multi-Target Tracking Algorithms," IEEE Trans.

Electronically approved by Ing. Magda Friedjungová, Ph.D. on 31 January 2024 in Prague.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Signal Process., vol. 68, pp. 3917–3928, 2020, doi: 10.1109/TSP.2020.3005309.
[2] R. Mahler, Advances in Statistical Multisource-Multitarget Information Fusion. Artech house, 2014.
[3] B. N. Vo and W. K. Ma, “The Gaussian mixture probability hypothesis density filter,” IEEE Transactions on Signal Processing, vol. 54, no. 11, pp. 4091–4104, 2006, doi: 10.1109/TSP.2006.881190.
[4] L. Stone, R. Streit, T. Corwin, and K. Bell, Bayesian Multiple Target Tracking. Artech house, 2013.
[5] R. R. Sanaga, “Multi-target tracking with uncertainty in the probability of detection,” MSc. Thesis, Purdue Univ., 2019.

Master's thesis

ESTIMATION OF DETECTION PROBABILITY IN MULTITARGET FILTERS USING OBJECT ADVANCED IMAGE PROCESSING TECHNIQUES

Bc. Michal Seibert

Faculty of Information Technology
Katedra aplikované matematiky
Supervisor: doc. Ing. Dedecius Kamil, Ph.D.
March 31, 2024

Citation of this thesis: Seibert Michal. *Estimation of detection probability in multitarget filters using object advanced image processing techniques.* Master's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2024.

Contents

List of Figures

List of Tables

List of code listings

Chtěl bych poděkovat především sit amet, consectetuer adipiscing elit. Curabitur sagittis hendrerit ante. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Cras pede libero, dapibus nec, pretium sit amet, tempor quis. Sed vel lectus. Donec odio tempus molestie, porttitor ut, iaculis quis, sem. Suspendisse sagittis ultrices augue.

Declaration

FILL IN ACCORDING TO THE INSTRUCTIONS. VYPLŇTE V SOULADU S POKYNY. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Curabitur sagittis hendrerit ante. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Cras pede libero, dapibus nec, pretium sit amet, tempor quis. Sed vel lectus. Donec odio tempus molestie, porttitor ut, iaculis quis, sem. Suspendisse sagittis ultrices augue. Donec ipsum massa, ullamcorper in, auctor et, scelerisque sed, est. In sem justo, commodo ut, suscipit at, pharetra vitae, orci. Pellentesque pretium lectus id turpis.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Curabitur sagittis hendrerit ante. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Cras pede libero, dapibus nec, pretium sit amet, tempor quis. Sed vel lectus. Donec odio tempus molestie, porttitor ut, iaculis quis, sem. Suspendisse sagittis ultrices augue. Donec ipsum massa, ullamcorper in, auctor et, scelerisque sed, est. In sem justo, commodo ut, suscipit at, pharetra vitae, orci. Pellentesque pretium lectus id turpis.

In Praze on March 31, 2024

Abstract

Fill in abstract of this thesis in English language. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Cras pede libero, dapibus nec, pretium sit amet, tempor quis. Sed vel lectus. Donec odio tempus molestie, porttitor ut, iaculis quis, sem. Suspendisse sagittis ultrices augue.

Keywords enter, comma, separated, list, of, keywords, in, ENGLISH

Abstrakt

Fill in abstract of this thesis in Czech language. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Cras pede libero, dapibus nec, pretium sit amet, tempor quis. Sed vel lectus. Donec odio tempus molestie, porttitor ut, iaculis quis, sem. Suspendisse sagittis ultrices augue.

Klíčová slova enter, comma, separated, list, of, keywords, in, CZECH

Seznam zkratek

DFA	Deterministic Finite Automaton
FA	Finite Automaton
LPS	Labelled Prüfer Sequence
NFA	Nondeterministic Finite Automaton
NPS	Numbered Prüfer Sequence
XML	Extensible Markup Language
XPath	XML Path Language
XSLT	eXtensible Stylesheet Language Transformations
W3C	World Wide Web Consortium

Introduction

Chapter 1
Thesis establishment

Multi-target tracking (MTT), a fundamental aspect of surveillance and monitoring systems, has undergone significant advancements in recent years, transforming it into a critical field with diverse applications across various domains. The scope of MTT extends beyond mere tracking, encompassing tasks such as object detection, identification, and trajectory prediction. The primary goal is to maintain a comprehensive situational awareness, providing invaluable information for decision-making processes in various applications, ranging from defense and surveillance to autonomous systems and robotics. This section provides an overview of the evolution, applications, significance, and current research focus of multi-target tracking.

1.1 Evolution and development

The roots of MTT can be traced back to the early 20th century when radar technology emerged during World War II [1]. Initially developed for single-target detection, radar systems laid the groundwork for subsequent advancements in multi-target tracking. As scenarios evolved and became more complex, the need for advanced tracking capabilities grew, prompting the development of more sophisticated algorithms.

The 1970s and 1980s witnessed the emergence of basic tracking algorithms, marking the initial forays into the field. Subsequent decades saw the integration of probabilistic techniques, such as the Kalman filter [2], which significantly enhanced tracking accuracy. . The 2000s marked the transition to data-driven approaches, with particle filters gaining popularity due to their ability to handle non-linear and non-Gaussian tracking scenarios [3]. However, due to the high computational complexity of particle filters, much attention is paid to data association filters such as JPDA [4] or RFS based filters [5].

1.2 Applications of multi-target tracking

The versatility of MTT is reflected in its diverse applications across various domains. In defense, MTT plays a pivotal role in monitoring and tracking multiple targets simultaneously, aiding in threat assessment, target prioritization, and distinguishing friend from foe.

The advent of autonomous systems, particularly in vehicles, has heightened the importance of MTT in predicting and tracking the movements of pedestrians, vehicles, and other obstacles. This application enhances the safety and efficiency of autonomous vehicles by providing real-time awareness of the surrounding environment [6].

Surveillance systems rely on multi-target tracking for monitoring activities in crowded environments and identifying suspicious behavior

Moreover, in fields such as robotics, defense, healthcare monitoring, and wildlife conservation, multi-target tracking systems contribute significantly to enhancing situational awareness, enabling real-time decision-making, improving resource allocation efficiency, and supporting various mission-critical tasks.

1.3 Multi target algorithms

The field of multi-target tracking is marked by a rich and diverse landscape of algorithms, each tailored to address specific challenges inherent in tracking multiple objects. These algorithms can be broadly categorized into association-based methods and Random Finite Set (RFS) based methods, each offering unique advantages and trade-offs.

Association-based methods form a foundational category in multi-target tracking, emphasizing the linking of measurements to existing tracks or the creation of new tracks. The well-established Kalman filter and its variants, such as the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF), fall under this category. More advanced methods considering clutter such as Probabilistic Association (PDA) filter, Joint Probabilistic Association (JPDA) filter or Multiple Hypothesis Tracker (MHT) filter are another examples of filters from this category.

In contrast to association-based methods, Random Finite Set (RFS) based methods provide a probabilistic framework to model multiple target states simultaneously. These methods operate on sets of possible target states, allowing for a more comprehensive representation of uncertainty and variability in tracking scenarios. The Probability Hypothesis Density (PHD) filter, Cardinalized Probability Hypothesis Density (CPHD) or Poisson multi-Bernoulli mixture (PMBM) filter are prominent examples of an RFS-based approach.

1.4 Research interests

Contemporary research in multi-target tracking is characterized by a strong emphasis on addressing key challenges to further enhance the performance and robustness of tracking systems. Current research focus deald with problems such as:

- **Data Association in complex environment:** Handling scenarios with high target density, occlusions, and clutter remains a complex issue. Robust methods for accurate and efficient data association in crowded and dynamic environments are still actively researched.
- **Handling non-linear and non-gaussian dynamics:** Real-world scenarios often exhibit non-linear and non-Gaussian characteristics. Improving tracking algorithms to effectively handle these complexities, possibly through the integration of advanced probabilistic models is an ongoing area of research.
- **Real-time processing and computational efficiency:** Many MTT algorithms, especially those with high computational demands, face challenges in meeting real-time processing requirements. Efficient algorithms that strike a balance between computational complexity and tracking accuracy are continuously sought.
- **Sensor fusion and heterogeneous data integration:** Integrating information from various sensors, each with its own characteristics and limitations, is an ongoing challenge. Developing robust methods for sensor fusion to improve tracking accuracy and reliability is an active research area.
- **Handling variability in target behavior:** Targets in real-world scenarios may exhibit diverse and unpredictable behaviors. Adapting tracking algorithms to handle varying target speeds, accelerations, and maneuvers remains an unsolved problem.

- **Online learning and adaptive algorithms:** Designing algorithms that can adapt and learn online as they encounter new scenarios or dynamic changes in the environment is a topic of interest. Adaptive tracking systems that can continuously improve their performance without extensive retraining are sought after.

As we will see later in this work, all Gaussian filters considering clutter rely on accurate setting of certain parameters like motion model or measurement model. Moreover, recursion requires estimation of parameters survival probability and detection probability. This thesis focuses on tracking targets in video data and takes advantage of image processing model YOLO (You Only Look Once). Due to this combination, we are able to estimate detection probability in time and enhance the tracking capability of the filter.

The probabilistic formulation of the RFS-based filters and the inherent Bayesian processing of available information allow to accommodate the uncertainty arising from the presence of false detections, missed detections, and data association ambiguities. Nevertheless, a fundamental challenge persists: The performance of the filters is highly sensitive to the accurate setting of the target detection probability. This quantity, representing the likelihood of correctly identifying and associating observations with actual targets, is a critical parameter. It has a substantial impact on the Bayesian updating of the prior information. However, in real-world scenarios, the sensor performance is susceptible to various environmental conditions. Adverse weather, occlusions, or just the nature of the current scenario can lead to variations in detection probabilities. A mismatch between assumed and actual detection probabilities can result in a suboptimal tracking performance, leading to missed detections, false alarms, or inaccurate target state estimates [7].

The RFS-based formulation of the PHD or (P)MBM filters naturally takes the uncertainty about the detection probability into account [7]. The update formulae involve it as a function of the target state. In the figurative sense, this allows to model it as a function of the spatial and temporal properties of the environment. Still, two difficulties arise. First, the (Gaussian) filters are analytically tractable only if the detection probabilities are scalar numbers. Second, the nature of the detection probabilities differs from scenario to scenario.

In general, several methods have been proposed to deal with unknown detection probabilities. A Gaussian-beta modeling of a slowly-varying detection profile in the Cardinalized PHD filter is reported in [8]; its alternative for MBM filters follows in [9], and for PMBM filter in [10]. In [11], the authors propose to overcome some deficiencies in the CPHD filter [8] by different clutter/detection probability models. Another variant was recently proposed in [12]. A track-state augmentation with an amplitude offset-based prediction of the detection probability appeared in [13], however, this method suffers difficulties in multistatic fields. An automatic identification system-based sensor performance assessment for clutter-free environments is developed in [14]. A recent paper [15] deals with the unknown detection profile in the trajectory PHD/CPHD filters. There, the algorithm learns from the history of the unknown target detection probability.

As we track object in video data, it allows us to avoid the generic solutions and focus on the peculiarities associated with this specific data type. In particular, advantage of YOLO is taken, offering high-performance real-time object detection with high accuracy and efficiency [16]. Its ability to simultaneously predict multiple bounding boxes and class probabilities within an image is used, providing a streamlined and efficient approach to object detection. However, YOLO-based multi-target tracking systems can still be compromised under conditions such as adverse weather, low lighting, or scenarios with occlusions. Sensors may encounter difficulties in accurately detecting and localizing targets, leading to gaps or errors in the tracking process.

1.5 Structure

The structure of this thesis is as follows:

Chapter 1

Chapter 2

Chapter 3

Chapter 4

Chapter 2
Background

Multi-target tracking relies heavily on mathematical concepts from probability theory, statistics, and linear algebra to model and infer the state of multiple objects over time. Key mathematical concepts involved in MTT include Bayesian inference, which provides a framework for updating estimates based on observed evidence, and the use of probabilistic models such as the Gaussian distribution and its extensions, such as Gaussian mixture models. In addition, MTT commonly employs state space models to represent the dynamics of object motion, with popular models including the constant velocity and constant acceleration models. These models are integrated into Bayesian filters, such as the Kalman filter, which recursively estimates the state of multiple targets given noisy measurements.

2.1 Bayesian inference

Bayesian inference stands as a foundational pillar within the realm of probabilistic reasoning, offering a fundamental methodology for systematically updating beliefs in response to observed evidence. At its heart stands Bayes' rule, a fundamental theorem in probability theory, that formalizes the process of revising prior beliefs in light of new data. The essence of Bayesian inference transcends mere statistical calculations; it embodies a philosophical stance towards uncertainty, emphasizing the incorporation of prior knowledge and the iterative refinement of beliefs through the assimilation of empirical observations. Within the context of multi-target tracking, Bayesian inference assumes a paramount role, providing a principled framework for joining information from disparate sources, such as sensor measurements, historical data, and domain expertise. By embracing Bayesian principles, MTT algorithms gain the capacity to model and quantify uncertainty inherent in tracking scenarios, thereby fostering robustness and adaptability in the face of dynamic and complex environments. Moreover, Bayesian inference empowers MTT systems to exploit contextual cues and domain-specific knowledge, enhancing their ability to discern meaningful patterns among noise and uncertainty. Due to the Bayesian inference, MTT researchers and practitioners are equipped with a powerful tool for navigating the intricacies of multi-target tracking, facilitating informed decision-making and advancing the frontier of robust systems. Thus, Bayesian inference emerges not merely as a mathematical construct but as a guiding philosophy underpinning the quest for understanding and reasoning in the face of uncertainty.

2.2 Bayes' rule

Bayes' rule, a cornerstone of Bayesian inference, embodies a fundamental principle in probability theory that underpins the systematic revision of beliefs in the face of new evidence. Mathematically expressed as a simple formula, Bayes' rule encapsulates the process of updating prior probabilities based on observed data, thereby yielding posterior probabilities that reflect the incorporation of new information. At its basis, Bayes' rule provides a formal mechanism for quantifying the impact of new evidence on the likelihood of various hypotheses or states of nature. The rule states that the posterior probability of a hypothesis given observed data is proportional to the product of the likelihood of the data given the hypothesis and the prior probability of the hypothesis, divided by the marginal likelihood of the data. In essence, Bayes' rule facilitates a principled approach to inference, allowing practitioners to integrate prior knowledge with empirical observations to arrive at more informed and reliable conclusions. In the context of multi-target tracking, Bayes' rule serves as the base upon which tracking algorithms are built, enabling the continuous refinement of estimates about the state of multiple targets based on sensor measurements and historical data. By adhering to the principles of Bayes' rule, MTT systems can effectively navigate the inherent uncertainty and complexity of tracking scenarios, thereby enhancing their robustness and accuracy.

▶ **Definition 2.1** (Conditional distribution)**.** *Let X and Y be jointly continuous random variables, f_Y continuous at y and $f_Y(y) > 0$. Then the conditional distribution function of X, given condition $Y = y$ is defined by*

$$F_{X|Y}(x|y) := lim_{\epsilon \to 0} \mathbf{P}\{X \leq x | Y \in (y, y+\epsilon)\} = \frac{\partial F(x,y)/\partial y}{f_Y(y)}. \tag{2.1}$$

Differentiating this, the conditional density function of X, given the condition $Y = y$ is

$$f_{X|Y}(x|y) := \frac{\partial^2 F(x,y)/[\partial x \partial y]}{f_Y(y)} = \frac{f(x,y)}{f_Y(y)}. \tag{2.2}$$

Naturally, fixing y, $F_{X|Y}(\cdot|y)$ and $f_{X|Y}(\cdot|y)$ are proper distribution and density functions, respectively. It is also clear that X and Y are independent if and only if $F_{X|Y}(x|y) = F_X(x)$ and $f_{X|Y}(x|y) = f_X(x)$ for all x and y for which these quantities are defined.

▶ **Definition 2.2** (Bayes' theorem)**.** *Let x and x be random variables with densities $f(y|x)$ and $f(x)$. Then Bayes' theorem is defined as*

$$f(x|y) = \frac{f(y|x)f(x)}{f(y)}, \quad f(y) > 0, \tag{2.3}$$

where

- $f(x|y)$ is the conditional posterior density of x,
- $f(x)$ is prior density,
- $f(y|x)$ is likelihood and $f(x)$ is marginal density of X, also called *evidence*, and is given by

$$f(y) = \int f(x,y)\partial x. \tag{2.4}$$

Combining equations 2.1, 2.2 and 2.4 to formula 2.3 we get complete formula for Bayes' theorem

$$f(x|y) = \frac{f(y|x)f(x)}{f(y)} = \frac{f(y|x)f(x)}{\int f(y|x)f(x)\partial x}. \tag{2.5}$$

The denominator is only the normalizing constant independent of x, often only proportionality is used

$$f(x|y) \propto f(y|x)f(x). \tag{2.6}$$

2.3 Multivariate Gaussian distribution

In the realm of multi-target tracking, various probability distributions are employed to model the uncertainty associated with target states and measurements. However, among these distributions, the multivariate Gaussian distribution holds a preeminent position due to its versatility, mathematical tractability, and empirical relevance. While other distributions may capture specific aspects of target behavior or measurement noise, the Gaussian distribution emerges as the cornerstone of MTT due to its ability to characterize complex probability distributions in multi-dimensional spaces. As a result common assumption among MTT filters is, that the targets follows linear Gaussian dynamic and measuremets models, as in [1], [17] or [18]. The Gaussian distribution finds ubiquitous application across various components of tracking algorithms, including

- **State representation:** Gaussian distributions are used to model the probability distributions of target states, allowing for efficient representation and propagation of uncertainty over time.
- **Measurement model:** Gaussian distributions are employed to model the likelihood of sensor measurements given the true target state, facilitating the incorporation of sensor data into the tracking process.
- **Filtering algorithms:** Gaussian-based filters leverage the Gaussian assumption to derive recursive estimation algorithms for tracking multiple targets with optimal efficiency and accuracy.
- **Data association:** Gaussian mixture models (GMMs), which represent mixtures of Gaussian distributions, are utilized for probabilistic data association in MTT, enabling robust handling of measurement uncertainty and target ambiguity.

▶ Note 2.3. The multivariate Gaussian distribution is a generalization of the univariate Gaussian distribution, but instead of a scalar mean and variace, there are mean vector and covariance matrix, that describes correlations between variables. The distribution of a $k-$dimensional random vector $\mathbf{X} = (X_1, \dots, X_k)^T$ and can be written in the notation

$$\mathbf{X} \sim \mathcal{N}(\mu, \mathbf{\Sigma}),$$

with $k-$dimensional mean vector $\mu = \mathsf{E}[\mathbf{X}] = (\mathsf{E}[X_1], \mathsf{E}[X_1], \dots, \mathsf{E}[X_k])^T$ and $k \times k$ positive semi-definite covariance matrix $\Sigma_{ij} = \mathsf{E}[(X_i - \mu_i)(X_j - \mu_j)] = Cov[X_i, X_j]$

The probability density function (PDF) is given by formula

$$\mathcal{N}(\mathbf{x}; \mu, \mathbf{\Sigma}) = \frac{1}{\sqrt{(2\pi)^k |\mathbf{\Sigma}|}} \exp(-\frac{1}{2}(\mathbf{x} - \mu)^T \mathbf{\Sigma}^{-1}(\mathbf{x} - \mu)), \tag{2.7}$$

where k is the length of the vector $\mathbf{x}$ and $|\cdot|$ denotes the determinant of a matrix. It should be noted that the exponent is known as Mahalanobis distance.

▶ **Definition 2.4** (Mahalanobis distance)**.** *For vectors x and y and a positive semi-definite matrix S, the Mahalanobis distance between two objects is defined as*

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T S^{-1} (\mathbf{x} - \mathbf{y})} \tag{2.8}$$

The Mahalanobis distance (MD) is the distance between two points in multivariate space and unlike Eucledian distance, it measures distances even between correlated points for multiple variables.

Target tracking filters regularly use conditional probabilities and joint distributions, especially Gaussian.

▶ **Theorem 2.5** (Conditional joint Gaussian distribution). *Let* $\mathbf{x}$ *and* $\mathbf{y}$ *are Gaussian random variables with distributions* $\mathcal{N}(\mathbf{x}; \mu_{\mathbf{x}}, \mathbf{\Sigma_{xx}})$ *and* $\mathcal{N}(\mathbf{y}; \mu_{\mathbf{y}}, \mathbf{\Sigma_{y}})$*, respectively. Let their joint probability is given by*

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix}; \begin{bmatrix}\mu_{\mathbf{x}}\\ \mu_{\mathbf{y}}\end{bmatrix}, \begin{bmatrix}\mathbf{\Sigma_{xx}} & \mathbf{\Sigma_{xy}}\\ \mathbf{\Sigma_{xy}} & \mathbf{\Sigma_{yy}}\end{bmatrix}\right). \tag{2.9}$$

Then the conditional distribution of $\mathbf{x}$ *given by* $\mathbf{y}$ *is defined as*

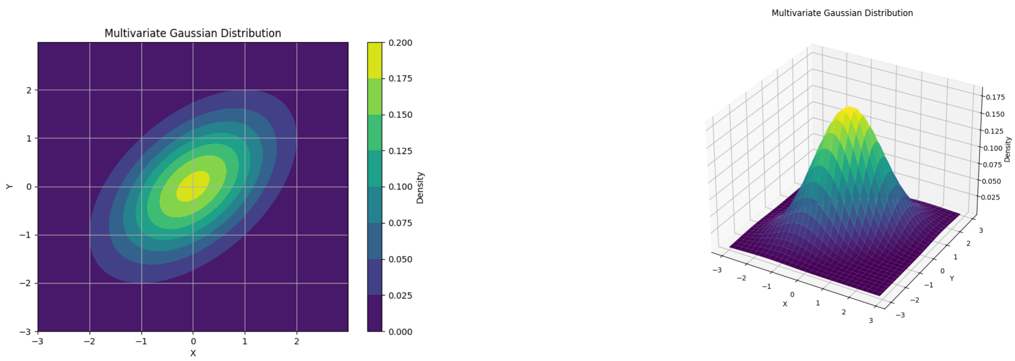
$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\mathbf{x}; \mu_{\mathbf{x}|\mathbf{y}}, \mathbf{\Sigma}_{\mathbf{x}|\mathbf{y}}), \tag{2.10}$$

where

$$\mu_{\mathbf{x}|\mathbf{y}} = \mu_{\mathbf{x}} + \mathbf{\Sigma_{xy}}\mathbf{\Sigma_{yy}}^{-1}(\mathbf{y} - \mu_{\mathbf{y}}), \tag{2.11}$$

$$\mathbf{\Sigma}_{\mathbf{x}|\mathbf{y}} = \mathbf{\Sigma_{xx}} - \mathbf{\Sigma_{xy}}\mathbf{\Sigma_{yy}^{-1}}\mathbf{\Sigma_{xy}^{T}}. \tag{2.12}$$

Examples of multivariate Gaussian distribution are shown in figure 2.1.



(a) Example of multivariate Gaussian distribution - contour plot.

(b) Example of multivariate Gaussian distribution - 3D plot.

Figure 2.1 Examples of plots of multivariate Gaussian distribution.

2.4 Gaussian mixture

Gaussian Mixture Models offer a flexible and powerful framework for representing complex probability distributions by combining multiple Gaussian components. In a Gaussian mixture model, a vector of parameters (e.g., observations of a signal) is modeled using a mixture distribution comprising several Gaussian components.

$$p(\Theta) = \sum_{i=1}^{k} w_i \mathcal{N}(\mu_i, \mathbf{\Sigma}_i), \tag{2.13}$$

where the i^{th} vector component is characterized by normal distributions with weights w_i, means μ_i and covariance matrices $\mathbf{\Sigma}_i$.

In the context of multi-target tracking, Gaussian mixture models find utility in capturing the complex nature of target states and measurements. MTT often involves dealing with linear Gaussian models, where the posterior density is represented as a mixture of one or more Gaussian components. This representation enables MTT algorithms to probabilistically model uncertainties associated with target dynamics, sensor measurements, and data association.

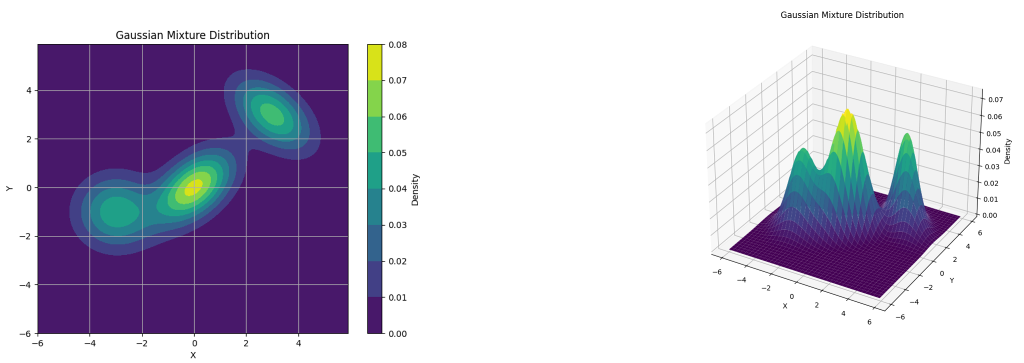
The probability density function (pdf) of a Gaussian mixture distribution is a simple sum of each Gaussian component, expressed as

$$p(x) = \sum_{i=1}^{k} w_i \mathcal{N}(\mathbf{x}; \mu_\mathbf{i}, \mathbf{\Sigma_i}), \tag{2.14}$$

where k is the number of Gaussian component, $w_i > 0$ is the weight of i^{th} component and $\sum_{i=1}^{k} w_i = 1$.

In MTT, Gaussian mixture models are particularly useful for tasks such as data association, where they probabilistically assign measurements to existing tracks or create new tracks based on the likelihood of observations given the target states. By modeling complex distributions of target states and measurements, Gaussian mixture models enable MTT algorithms to handle uncertainties and ambiguities inherent in real-world tracking scenarios, including occlusions, clutter, and target interactions.

Examples of multivariate Gaussian mixture distribution are shown in figure 2.2.



(a) Example of multivariate Gaussian mixture distribution - contour plot.

(b) Example of multivariate Gaussian mixture distribution - 3D plot.

Figure 2.2 Examples of plots of multivariate Gaussian mixture distribution. There are three components in figures with means $\{[0, 0], [3, 3], [-3, -1]\}$. Note that the density of peaks is very low, because the sum has to be equal 1.

2.5 State space model

State-space models serve as a fundamental framework for describing the evolution of dynamic systems over time. In the context of multi-target tracking, state-space models provide a formalism for representing motion dynamics of targets and the measurement process, enabling efficient and accurate inference of target states from sensor data.

At the core of a state-space model lies a set of latent variables, known as the state vector, which encapsulates the unobservable quantities of interest, such as the position, velocity, and acceleration of targets in MTT. The dynamics governing the evolution of the state vector are typically described by a transition model, which specifies how the state evolves over time according to a probabilistic process. This transition model can take various forms depending on the nature of the system dynamics, ranging from simple linear or nonlinear models to more complex stochastic processes.

In addition to the transition model, state-space models incorporate an observation model that describes the relationship between the observed measurements and the underlying state variables. This observation model accounts for the uncertainties and noise inherent in the measurement process, allowing for the probabilistic mapping of observed data to the latent state space.

State-space model can be represented as a pair of stochastic equations.

1. **State transition equation:**

$$\mathbf{x_k} = f(\mathbf{x_{k-1}}, \mathbf{u_k}, \mathbf{w_k}), \tag{2.15}$$

   where $\mathbf{x_k}$ represents the state vector at time k, f denotes the transition function describing the evolution of the state, $\mathbf{u_k}$ represents optional control inputs, and $\mathbf{w_k}$ denotes process noise.

2. **Observation equation:**

$$\mathbf{z_k} = h(\mathbf{x_k}, \mathbf{v_k}), \tag{2.16}$$

   where $\mathbf{z_k}$ represents the observed measurements at time k, h denotes the observation function mapping the state to measurements, and v_k represents the measurement noise.

In MTT, state-space models provide a natural framework for representing the motion dynamics of multiple targets and the sensor measurements associated with each target. Each target is typically associated with its own state vector, allowing for simultaneous tracking of multiple objects within the same probabilistic framework.

State-space models enable MTT algorithms to perform a range of tasks, including target prediction, data association, and state estimation by propagating the state forward in time using the transition model and updating the state based on observed measurements using the observation model. By incorporating uncertainty explicitly into the tracking process, state-space models facilitate robust inference of target states in complex and dynamic tracking scenarios.

While state-space models offer a powerful framework for multi-target tracking, they also present several challenges and considerations.

- **Model complexity:** Designing an appropriate state-space model requires careful consideration of the underlying dynamics and measurement process, which can be challenging in complex tracking scenarios with non-linearities and uncertainties.

- **Parameter estimation:** Estimating the parameters of a state-space model from data, such as the transition and observation matrices, can be computationally demanding and prone to issues such as overfitting or underfitting.

- **Computational Complexity:** Performing inference in state-space models often involves recursive algorithms such as the Kalman filter or particle filter, which can be computationally intensive, especially in high-dimensional or real-time tracking applications.

Despite these challenges, state-space models remain a cornerstone of multi-target tracking, offering a principled and flexible framework for representing and reasoning about dynamic systems in the presence of uncertainty. There are many state space models used in MTT. Before we define two of the most common ones in next sections, it should be noted, that definitions in (2.15) and (2.16) are too general, because f_k and h_k could by any functions.

To get a closed-form solution in the Bayesian inference framework, we need to choose conjugate distributions for the likelihood and the prior. The Kalman filter (see Section 2.8 for more) works for the Gaussian-linear case, where the functions f_t and h_t are linear and noisy variables and are distributed as Gaussians with zero mean, i.e., $w_k, v_k \sim \mathcal{N}(0, \Sigma)$. The Gaussian linear state space model has the formulation

$$p(x_k|x_{k-1}) = Fx_{k-1} + Bu_k + w_k \qquad w_k \sim \mathcal{N}(0, Q), \tag{2.17}$$
$$p(z_k|x_k) = Hx_k + v_k \qquad v_k \sim \mathcal{N}(0, R), \tag{2.18}$$
$$p(x_0) \sim \mathcal{N}(\hat{x}_0, P_0), \tag{2.19}$$

where

- F is the transition matrix of appropriate dimension,
- B is the input matrix of appropriate dimension,
- H is the measurement matrix of appropriate dimension,
- Q is the symmetric positive semi-definite matrix that desribes the statistical properties of motion noise w_k,
- R is the symmetric positive semi-definite matrix that desribes the statistical properties of measurement noise v_k,
- $\hat{x}_0$ and P_0 are the mean and the covariance matrix of the prior state.

The control variable u_k, in general, represents some input signal from the environment, and B specifies how the input signal affects the dynamic system. This variable is ususally not considered in MTT scenarios. In multi-target tracking, most often with Gaussian linear models is worked, thus following formulation is instead used.

$$p(x_k|x_{k-1}) = \mathcal{N}(x_k; Fx_{k-1}, Q), \tag{2.20}$$
$$p(z_k|x_k) = \mathcal{N}(z_k; Hx_k, R), \tag{2.21}$$
$$p(x_0) \sim \mathcal{N}(x_0; \hat{x}_o, P_0). \tag{2.22}$$

2.5.1 Constant velocity model

The constant velocity model (CVM) is a fundamental component of multi-target tracking systems, providing a simplified yet effective representation of target motion dynamics over time. This model assumes that the target's velocity remains constant between consecutive time steps, making it particularly suitable for tracking objects with relatively smooth and predictable motion patterns. In this section, we explore the conceptual basis, mathematical formulation, and practical implications of the constant velocity model in the context of MTT.

At its core, the constant velocity model embodies the notion of inertia, where a target maintains a constant velocity unless acted upon by external forces. This conceptual simplicity allows for a straightforward representation of target motion, making the constant velocity model a popular choice for MTT applications where targets exhibit relatively uniform and predictable movement behaviors.

Mathematically, the constant velocity model describes the evolution of a target's state vector over time in terms of its position and velocity. At each time step k, the state vector $\mathbf{x}_k$ comprises the position $[x_{1,k}, x_{2,k}]$ and velocity v of the target

$$\mathbf{x}_k = [x_{1,k}, x_{2,k}, v_{1,k}, v_{2,k}]^T. \tag{2.23}$$

The dynamics of the constant velocity model can be represented using a state transition matrix F and a process noise vector w_k, where the state transition matrix captures the relationship between the target's state at consecutive time steps

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{w}_k. \tag{2.24}$$

All together, vector $\mathbf{x}_k$, matrices $\mathbf{F}$ and $\mathbf{Q}$, might, for example, be represented as

$$\mathbf{x}_k = \begin{bmatrix} x_{1,k} \\ x_{2,k} \\ v_{1,k} \\ v_{2,k} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{Q} = q^2 \begin{bmatrix} \frac{dt^3}{3} & 0 & \frac{dt^2}{2} & 0 \\ 0 & \frac{dt^3}{3} & 0 & \frac{dt^2}{2} \\ \frac{dt^2}{2} & 0 & dt & 0 \\ 0 & \frac{dt^2}{2} & 0 & dt \end{bmatrix}, \tag{2.25}$$

where dt stands for delta time, i.e., elapsed time between the last estimation and the current one, q is the motion noise parameter, which represents the uncertainty in the state transition. The measurement model transforms a state vector from the state space into the measurement space. Since filters derived from Kalman filter assumes only linear models, the measurement model for CVM assumes the same space as in the state vectors. As a result, the observation matrix $\mathbf{H}$ and the noise matrix $\mathbf{R}$ can be, with respect to (2.25), formulated

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad \mathbf{R} = r^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{2.26}$$

where r determines the variance of the measurement noise.

To obtain optimal results given by filters, it is essential to set parameters r and q appropriately. There are three main ways to do it. The first one requires the knowledge of motion noise given by tracked object and the knowledge of sensor's error when detecting targets, i.e. the noise measurement. This method is often used in simulations and experiments. The second one is simple trial and error method with reasonable choices. The last one is using automated methods, like in [19].

In the context of MTT, the constant velocity model serves as a fundamental building block for tracking algorithms, providing a simplified yet effective representation of a target motion. By assuming the constant velocity, MTT algorithms can predict the future positions of targets based on their current state and velocity, facilitating trajectory estimation and target prediction.

One common application of the constant velocity model in MTT is in the design of prediction algorithms, where future positions of targets are estimated based on their current state and velocity information. These predictions are essential for anticipating target movements and facilitating data association, enabling MTT algorithms to maintain track continuity and adapt to changes in target behavior over time.

Moreover, the constant velocity model can be seamlessly integrated into recursive Bayesian filters, such as the Kalman filter, for state estimation in MTT. By incorporating the constant velocity model into the state-space representation of the tracking problem, Kalman filter-based algorithms can effectively fuse measurement information with dynamic predictions to estimate the most likely trajectories of targets over time.

2.5.2 Constant acceleration model

The Constant Acceleration Model (CAM) stands as a sophisticated extension of the state-space model in multi-target tracking, offering a more comprehensive representation of target motion dynamics. Unlike simpler models such as the Constant Velocity Model, which assumes a constant velocity for targets over time, the CAM acknowledges the potential for changes in target acceleration, allowing for more accurate and flexible trajectory estimation. This section delves into the conceptual basis, mathematical formulation, and practical implications of the Constant Acceleration Model in the context of MTT.

In MTT scenarios, targets often exhibit varying degrees of acceleration due to factors such as changes in speed, direction, or environmental influences. Ignoring these accelerative effects can lead to biased trajectory estimates and diminished tracking accuracy. The CAM addresses this limitation by incorporating an additional acceleration component into the state-space model, enabling more faithful representation of target motion dynamics. By accounting for changes in acceleration, the CAM provides MTT algorithms with greater flexibility and adaptability in tracking targets with non-uniform motion profiles.

Mathematically, the Constant Acceleration Model extends the state transition function of the state-space model to accommodate changes in acceleration over time. At each time step k, the evolution of the target's state vector x_k is governed by a set of dynamic equations that describe the position, velocity, and acceleration of the target

$$\mathbf{x}_k = [x_{1,k}, x_{2,k}, v_{1,k}, v_{2,k}, a_{1,k}, a_{2,k}]^T, \tag{2.27}$$

where, as in CVM, $x_{1,k}$, $x_{2,k}$ is the position of an target in two-dimensional space, $v_{1,k}, v_{2,k}$ is the velocity and $a_{1,k}, a_{2,k}$ are the accelerations in both directions. Matrices F and Q can then be expressed

$$\mathbf{x}_k = \begin{bmatrix} x_{1,k} \\ x_{2,k} \\ v_{1,k} \\ v_{2,k} \\ a_{1,k} \\ a_{2,k} \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} 1 & 0 & dt & 0 & \frac{1}{2}dt^2 & 0 \\ 0 & 1 & 0 & dt & 0 & \frac{1}{2}dt^2 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{Q} = q^2 \begin{bmatrix} \frac{dt^4}{4} & 0 & \frac{dt^3}{3} & 0 & \frac{dt^2}{2} & 0 \\ 0 & \frac{dt^4}{4} & 0 & \frac{dt^3}{3} & 0 & \frac{dt^2}{2} \\ \frac{dt^3}{3} & 0 & \frac{dt^2}{2} & 0 & dt & 0 \\ 0 & \frac{dt^3}{3} & 0 & \frac{dt^2}{2} & 0 & dt \\ \frac{dt^3}{3} & 0 & \frac{dt^2}{2} & 0 & dt & 0 \\ 0 & \frac{dt^3}{3} & 0 & \frac{dt^2}{2} & 0 & dt \end{bmatrix}. \tag{2.28}$$

The measurement model remains unchanged from the standard state-space model, relating the observed measurements $\mathbf{z}_k$ to the target's true state $\mathbf{x}_k$ through a measurement function $h(\cdot)$ corrupted by measurement noise v_t

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{R} = r^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{2.29}$$

The Constant Acceleration Model offers several practical advantages in MTT applications:

- Improved trajectory estimation: By accounting for changes in acceleration, the CAM enables more accurate and realistic trajectory estimation, particularly for targets exhibiting non-uniform motion patterns or sudden changes in velocity.
- Enhanced predictive capability: The inclusion of acceleration dynamics allows MTT algorithms to make more informed predictions about future target states, improving tracking performance and reducing prediction errors.
- Robustness to dynamic environments: In dynamic environments with varying levels of congestion, obstacles, or traffic patterns, the CAM provides MTT algorithms with greater robustness and adaptability, enabling effective tracking even in challenging scenarios.
- Applications in autonomous systems: In autonomous systems such as self-driving cars, drones, or robotics, the CAM plays a crucial role in motion planning, obstacle avoidance, and trajectory prediction, enhancing the safety and efficiency of autonomous navigation.

In summary, the Constant Acceleration Model represents a significant advancement in multi-target tracking, offering a more nuanced and comprehensive approach to modeling target motion dynamics. By incorporating acceleration effects into the tracking process, the CAM enables MTT algorithms to achieve higher accuracy, robustness, and predictive capability, making it an indispensable tool for a wide range of applications.

2.6 Hidden Markov Model

In the domain of multi-target tracking, the Hidden Markov Model (HMM) serves as a framework for capturing the temporal dynamics of target behavior in complex environments. Building upon the principles of Markov processes, the HMM extends the state-space model by introducing hidden states that govern the evolution of observable measurements over time. Markov process of the first order is a model, in which current state depends only on the previous state

$$p(x_k|x_1,\dots,x_{k-2},x_{k-1},z_1,\dots,z_{k-2},z_{k-1}) = p(x_k|x_{k-1}) \quad \text{(transition probability)}, \tag{2.30}$$

$$p(z_k|x_1,\dots,x_{k-2},x_{k-1},x_k,z_1,\dots,z_{k-2},z_{k-1}) = p(z_k|x_k) \quad \text{(observation likelihood)}, \tag{2.31}$$

$$p(x_0) \quad \text{(initial state)}. \tag{2.32}$$

In models of higher order, the transition probability is

$$p(x_k|x_{k-1},\dots,x_{k-n}), \tag{2.33}$$

where n is the number of the order. Details of this model is not described in this work and can be seen in [20].

In cases where the process is not didectly observable, but can be observed through another observable variable z_k, we often talk about Hidden Markov process.

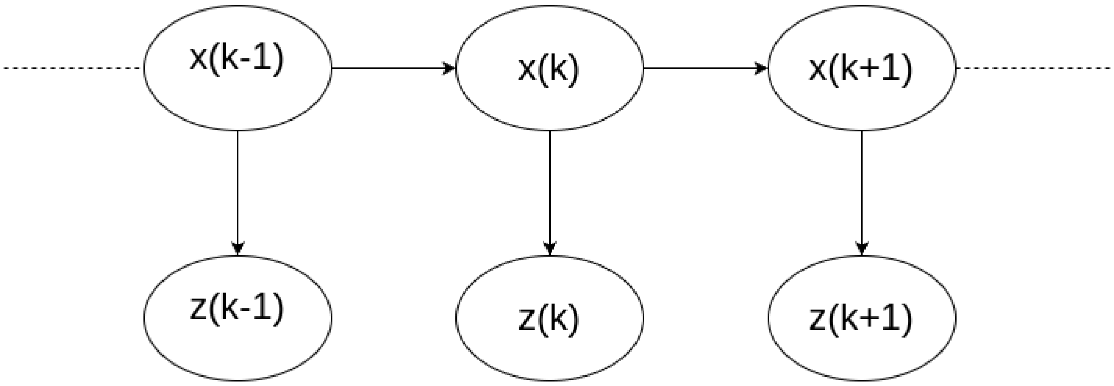


Figure 2.3 Demonstration of the course of the hidden Markov process.

At its core, the Hidden Markov Process represents a stochastic process characterized by a set of hidden states that transition probabilistically over time. While these hidden states are unobservable, they influence the generation of observable measurements, which are assumed to be conditionally independent given the hidden states. In the context of MTT, the hidden states may represent latent attributes of targets, such as their locations, velocities, or motion patterns, while the observable measurements correspond to sensor readings or detections obtained from surveillance systems.

2.7 Bayes' filter

The Bayes filter is a recursive framework that estimates an internal state of the system over time using measurements. The Bayes filter leverages both motion and measurement models, operating under the assumption that these models accurately describe the system's behavior. Operating within a recursive framework, the Bayes filter continually estimates the internal state of the system over time using available measurements. Each iteration of the filter entails two fundamental steps: prediction and update. During the prediction step, the filter anticipates the internal state x_k based on the preceding state x_{k-1} and the motion model $p(x_k|x_{k-1})$. This prediction step is commonly referred to as the Chapman-Kolmogorov equation [21].

▶ **Theorem 2.6** (The Chapman-Kologorov equation prediction step). *Given the set of measurements $z_{0:k-1} = z_0, \ldots, z_{k-1}$, set of control variables $u_{0:k-1} = u_0, \ldots, u_{k-1}$ and current state x_{k-1} and the motion model $p(x_k|x_{k-1})$, the prediction step is as follows:*

$$p(x_k|z_{0:t-1}, u_{0:k}) = \int p(x_k|x_{k-1}, u_k)p(x_{k-1}|z_{0:k-1}, u_{0:k-1})dx_{k-1}, \tag{2.34}$$

where the integral is taken over the entire state space of x_{k-1}, and $p(x_{k-1}|z_{0:k-1}, u_{0:k-1})$ is the posterior density of the state at time $k-1$.

On the update step, the Bayes' filter corrects (updates) the prediction step with measurements z_k using the measurement model $p(z_k|x_k)$. This step uses common Bayes' rule.

▶ **Theorem 2.7** (The update step of Bayes' filter). *Given the output of Bayes' filter's prediction step, the measurements z_k observed at time k and the measurement model $p(z_k|x_k)$, the update step of Bayes' filter is formulated as follows:*

$$p(x_k|z_{0:t}, u_{0:t}) = \frac{p(z_k|x_k)p(x_k|z_{0:k-1}, u_{0:k})}{p(z_k|z_{0:k-1})} \propto p(z_k|x_k)p(x_k|z_{0:k-1}, u_{0:k}) \tag{2.35}$$

These two steps work in an iterative method, where, in each iteration, first prediction step is used to predict the state x_k and then update step is used to correct our guess with provided measurement.

2.8 Kalman filter

The Kalman filter, named after its developer Rudolf E. Kalman, has a rich history dating back to the early 1960s when Kalman first introduced the algorithm in a series of landmark papers. Initially developed for aerospace applications, the Kalman filter gained prominence for its ability to provide optimal state estimation in the presence of noisy measurements and uncertain dynamics.

Over the years, the Kalman filter has found widespread usage across diverse fields and industries, including aerospace, robotics, finance, and telecommunications. Its applications range from tracking spacecraft trajectories and guiding missiles to monitoring financial markets and controlling autonomous vehicles. The filter's versatility, efficiency, and effectiveness have made it a cornerstone of modern estimation and control systems.

The Kalman filter operates on the principles of recursive Bayesian estimation, utilizing a system dynamics model and noisy measurements to predict and update the state of a dynamic system over time. Its mathematical elegance and simplicity make it well-suited for real-time applications, enabling accurate and efficient state estimation even in complex and uncertain environments.

Despite its remarkable success, the Kalman filter continues to evolve, with extensions and variations such as the extended Kalman filter (EKF), unscented Kalman filter (UKF), and particle filter (PF) addressing more challenging scenarios involving non-linear dynamics and non-Gaussian noise.

In summary, the Kalman filter stands as a seminal contribution to the field of estimation and control, with a rich history of development and widespread usage across diverse domains. Its continued relevance and versatility underscore its status as a foundational tool for state estimation and tracking in modern technological applications.

2.8.1 Kalman filter inference

In section 2.5 we talked about State space model, which is used by Kalman filter. Thus we work with equations

$$x_k = Fx_{k-1} + w_t \tag{2.36}$$
$$z_t = Hx_k + v_k, \tag{2.37}$$

where both noise variables are independent and with zero mean

$$w_k \sim \mathcal{N}(0, Q), \tag{2.38}$$
$$v_k \sim \mathcal{N}(0, R). \tag{2.39}$$

Then the state and measurement distributions follows

$$x_k \sim \mathcal{N}(Fx_{k-1}, Q) \qquad \text{with density} \quad p(x_k|x_{k-1}), \tag{2.40}$$
$$z_k \sim \mathcal{N}(Hx_k, R) \qquad \text{with density} \quad p(z_k|x_k). \tag{2.41}$$

Because Kalman filter is bayesian, we need prior distribution for x_k. Model z_k is Gaussian, conjugated prior and is then also Gaussian with mean x^+_{k-1} and covariance matrix P^+_{k-1},

$$p(x_k|z_{0:k-1}) = \mathcal{N}(x^+_{k-1}, P^+_{k-1}). \tag{2.42}$$

The prediction step is derived from formula (2.34). Note, that by multiplicating two gaussian distributions we again get gaussian distribution $\mathcal{N}(x^-_k, P^-_k)$ with hyperparemeters

$$x^-_k = Fx^+_{k-1} \tag{2.43}$$
$$P^-_k = FP^+_{k-1}F^T + Q. \tag{2.44}$$

By multiplicating two normal distributions followed by marginalization, we enumerate the state equation. The estimation of state x^-_k is just applying model variables into appropriate equations. The estimated covariance P^-_k expresses the degree of uncertainty of the estimation. By applying this step, the uncertainty grows.

The update step corrects the prediction step by applying new observed measurements z_k. For that, formula (2.35) is used. The model is transformed into exponential form

$$\begin{aligned} p(z_k|x_k) &\propto \exp\left\{-\frac{1}{2}(z_k - Hx_k)^T R^{-1}(z_k - Hx_k)\right\} \\ &= \exp\left\{ Tr\left(\underbrace{-\frac{1}{2}\begin{bmatrix}-1\\ x_k\end{bmatrix}\begin{bmatrix}-1\\ x_k\end{bmatrix}^T}_{\eta} \underbrace{\begin{bmatrix}z_k^T\\ H^T\end{bmatrix} R^{-1} \begin{bmatrix}z_k^T\\ H^T\end{bmatrix}^T}_{T(z_k)} \right)\right\}. \end{aligned} \tag{2.45}$$

The conjugated shape has the form

$$\begin{aligned} p(x_k|z_{0:k-1}) &\propto \exp\left\{-\frac{1}{2}(x_k - x^-_k)^T (P^-_k)^{-1}(x_k - x^-_k)\right\} \\ &= \exp\left\{ Tr\left(\underbrace{-\frac{1}{2}\begin{bmatrix}-1\\ x_k\end{bmatrix}\begin{bmatrix}-1\\ x_k\end{bmatrix}^T}_{\eta} \underbrace{\begin{bmatrix}(x^-_k)^T\\ I\end{bmatrix} (P^-_k)^{-1} \begin{bmatrix}(x^-_k)^T\\ I\end{bmatrix}^T}_{\xi_k} \right)\right\}, \end{aligned} \tag{2.46}$$

where I is identity matrix of appropriate shape.

The bayesian update is a sum of the hyperparameter and sufficient statistic,

$$\begin{aligned}\xi_k &= \xi_{k-1} + T(z_k) \\ &= \begin{bmatrix} (x_k^-)^T (P_k^-)^{-1} x_k^- + z_k^T R^{-1} z_k, & (x_k^-)^T (P_k^-)^{-1} + z_k^T R^{-1} H \\ (P_k^-)^{-1} (x_k^-)^T + H^T R^{-1} z_k, & (P_k^-)^{-1} + H^T R^{-1} H. \end{bmatrix}\end{aligned} \tag{2.47}$$

The posterior parameters are then derived

$$\begin{aligned}P_k^+ &= (\xi_{k;[2,2]})^{-1} \\ &= \left[(P_k^-)^{-1} + H^T R^{-1} H\right]^{-1} \\ &= (I - K_k H) P_k^- \end{aligned} \tag{2.48}$$

$$\begin{aligned}x_k^+ &= (\xi_{k;[2,2]})^{-1} \xi_{k;[2,1]} \\ &= P_k^+ \left[(P_k^-)^{-1} (x_k^-)^T + H^T R^{-1} z_k\right] \\ &= x_k^- + P_k^+ H^T R^{-1} (z_k - H x_k^-), \end{aligned} \tag{2.49}$$

where

$$K_k = P_k^- H^T (R + H P_k^- H^T)^{-1} \tag{2.50}$$

is the Kalman gain. This form is the optimal Kalman gain, as it minimizes the root mean squared error. In general, the greater the gain, the greater the emphasis of the new measurements. The filter is then more sensitive, but filters noise less easily.

In literature it is common to write each equation in following or similar form

$$\hat{z}_k^- = H \hat{x}_k^-, \qquad \text{(measurement prediction)} \tag{2.51}$$

$$\nu_t = z_t - \hat{z}_k^-, \qquad \text{(innovation or prediction error } z_t\text{)} \tag{2.52}$$

$$S_t = H P_k^- H^T + R, \qquad \text{(covariance of innovation } \nu_t\text{)} \tag{2.53}$$

$$K_t = P_k^- H^T S_t^{-1}, \qquad \text{(Kalman gain)} \tag{2.54}$$

$$x_k^+ = \hat{x}_k^- + K_t \nu_t, \qquad \text{(aposteriori state estimation } x_t\text{)} \tag{2.55}$$

$$P_k^+ = (I - K_t H) P_k^-. \qquad \text{(aposteriori covariance)} \tag{2.56}$$

The pseudo algorithm of Kalman filter is shown in algorithm 1.

Algorithm 1 Kalman Filter Algorithm

Inputs: Initial state estimate $\hat{x}_0$, initial covariance matrix P_0, system dynamics matrix F_k, process noise covariance matrix Q_k, measurement matrix H_k, measurement noise covariance matrix R_k, and measurements Z.
Outputs: Updated state estimate $\hat{x}_k$ and updated covariance matrix P_k.

procedure INITIALIZATION(x_0, P)
    $\hat{x}_0 = x_0$ ▷ Initialize state estimate
    $P_0 = P$ ▷ Initialize error covariance matrix
end procedure

procedure PREDICTION STEP
    $\hat{x}_{k|k-1} = F_k \hat{x}_{k-1}$ ▷ Predict state estimate
    $P_{k|k-1} = F_k P_{k-1} F_k^T + Q_k$ ▷ Predict covariance:
end procedure

procedure UPDATE STEP(z_k)
    $K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1}$ ▷ Kalman gain
    $\hat{x}_k = \hat{x}_{k|k-1} + K_k(z_k - H_k \hat{x}_{k|k-1})$ ▷ Update state estimate
    $P_k = (I - K_k H_k) P_{k|k-1}$ ▷ Update covariance
end procedure

procedure KALMAN FILTER RECURSION
    Initialization
    for all $\mathrm{z}_k \in \mathrm{Z}$ **do**
        Perform Prediction Step
        Perform Update Step with z_k
    end for
end procedure

Chapter 3

Target tracking

Target tracking plays a pivotal role in radar systems, where the primary objective is to estimate the trajectory, position and other characteristics of moving objects within a surveillance area. Derived from the foundational principles of the Kalman filter, target tracking algorithms are tailored to the specific requirements and challenges inherent in radar applications.

Radar-based target tracking encounters various complexities, including survivability and the presence of multiple targets in the same neighbour. Survivability concerns the algorithm's ability to maintain accurate estimates despite target maneuvers, occlusions, or potential target loss. Furthermore, the phenomenon of multiple targets in the same neighbour introduces significant ambiguities and challenges in trajectory estimation and association.

In the realm of single-target tracking, algorithms such as the Probabilistic Data Association (PDA) filter and its derivatives, such as the Joint Probabilistic Data Association (JPDA) filter or the Integrated Probabilistic Data Association (IPDA) filter, are commonly employed. These algorithms provide robust solutions for associating measurements with existing tracks and updating target state estimates in dynamic scenarios.

Beyond single-target tracking, radar systems often necessitate multi-target tracking approaches. These approaches, particularly those based on Random Finite Sets (RFS), offer advanced capabilities for tracking multiple targets simultaneously. RFS-based filters, including the Probability Hypothesis Density (PHD) filter and the Multi-Bernoulli filter, excel in scenarios where the number of targets is uncertain or dynamic.

In this chapter, we delve into the intricacies of radar-based target tracking algorithms, exploring their theoretical foundations, practical implementations, and performance characteristics. By understanding the diverse range of tracking algorithms available and their respective strengths, we can design and deploy effective tracking systems to meet the demands of modern surveillance and reconnaissance applications.

3.1 Data association

Data association uncertainty arises in remote sensing systems, such as radar, sonar, or electro-optical devices, when measurements are obtained from sources that may not necessarily be the target of interest [22]. This uncertainty occurs particularly in situations where the target signal is weak, necessitating a lower detection threshold, which may result in the detection of background signals, sensor noise or clutter. Additionally, data association uncertainty can occur when multiple targets are present in close proximity. Utilizing spurious measurements in a tracking filter can lead to divergence of the estimation error and, consequently, track loss.

Addressing this challenge involves two primary problems. The first is the selection of appropriate measurements to update the state of the target of interest in the tracking filter, which can be a Kalman filter or an extended Kalman filter. The second problem involves determining whether the filter needs modifications to account for data association uncertainty. The objective is to obtain the minimum mean square error (MMSE) estimate of the target state and associated uncertainty.

The optimal estimator involves the recursive computation of the conditional probability density function (pdf) of the state, with detailed conditions provided under which this pdf serves as a sufficient statistic in the presence of data association uncertainty.

3.1.1 Validation region

In target tracking scenarios, the process of signal detection provides measurements, from which the appropriate ones for inclusion in the target state estimator are chosen. In radar systems, for instance, the signal reflected from the target of interest is sought within a specific time interval, determined by the expected range of the target when it reflects the transmitted energy. A range gate is established, and detections falling within this gate can be associated with the target of interest. These measurements could include informations such as range, azimuth, elevation, or direction cosines. For radar or active sonar, we might also measure how fast something is moving toward or away from us. For passive sonar, we might look at the direction something's coming from, when it arrives, and the difference in sounds it makes. And for optical sensors, we might measure the angle it is seen from. By setting up a multidimensional gate, the signal from the target is detected efficiently, avoiding the need to search for it across the entire measurement space.

However, while a measurement within the gate is a candidate for association with the target, it is not guaranteed to have originated from the target itself. Thus, the establishment of a validation region becomes necessary. The validation region is designed to ensure that the target measurement falls within it with a high probability, known as the gate probability, based on the statistical characterization of the predicted measurement. In the event, where more than one detection appears within the gate, association uncertainty arises. With this uncertainty, it is important to figure out which measurement really comes from the target and should be used to update our tracking information. This includes things like the target's estimated position and its variability, or more broadly, the key information we need about the target. Measurements outside the validation region can be disregarded, as they are too distant from the predicted measurement and are unlikely to have originated from the target of interest. This scenario typically arises when the gate probability is close to unity, and the statistical model used to define the gate is accurate.

3.2 Clutter

When it comes to clutter, two scenarios may occur. The first one is a single target in clutter. This problem of tracking a single target in clutter appears when several measurements appear in the validation region. The validated measurements comprise the accurate measurement, if detected within this region, along with spurious measurements originating from clutter or false alarms. In air traffic control systems, where cooperative targets are involved, each measurement includes a target identifier known as the squawk number. If this identifier is entirely reliable, data association uncertainty is eliminated. However, in cases where a potentially hostile target is non-cooperative, data association uncertainty becomes a significant challenge.

Figure 3.1 illustrates a scenario involving multiple validated measurements. The validation region depicted in the figure is two-dimensional and takes the form of an ellipse centered at the predicted measurement $\hat{z}$. The elliptical shape of the validation region arises from the assumption that the error in the target's predicted measurement, known as the innovation, follows a Gaussian distribution. The parameters defining the ellipse are determined by the covariance matrix S of the innovation.

All measurements within the validation region have the potential to originate from the target of interest, although only one of them is the true measurement. As a result, the possible association events include: z_1 originating from the target, with z_2 and z_3 being from clutter; z_2 originating from the target, with z_1 and z_3 being from clutter; z_3 originating from the target, with z_2 and z_1 being from clutter; or all measurements being from clutter. These association events are mutually exclusive and exhaustive, enabling the application of the total probability theorem to obtain the state estimate in the presence of data association uncertainty.

Under the assumption of a single target, the spurious measurements are considered random interference. A common model for such false measurements assumes that they are uniformly spatially distributed and independent across time, corresponding to residual clutter. Any constant clutter is assumed to have already been removed.

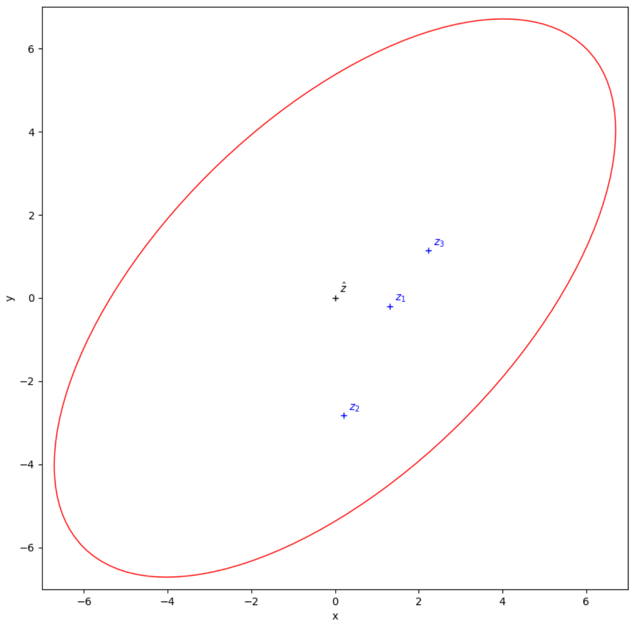


Figure 3.1 Several measurements z_i appeared in the validation region of a single target. $\hat{z}$ is a predicted measurement and none or any of the measurement $z_1 - z_3$ may have originated from the target.

When several targets as well as clutter or false alarms appear in the same neighbour, the data association becomes even more challenging. Figure 3.2 shows this scenario, where predicted measurement for the targets are close to each other. These predicted points are labeled as $\hat{z_1}$ and $\hat{z_2}$. In this figure many association combinations are possible; z_1 from target $\hat{z_1}$ or clutter; z_2 from target $\hat{z_1}$ or clutter; z_4 from target $\hat{z_2}$ or clutter; z_3 from target $\hat{z_1}$ or target $\hat{z_2}$ or clutter. However, if z_3 originated from target $\hat{z_1}$, then it is probable, that z_4 originated from target $\hat{z_2}$.

This scenario demonstrates the intricate relationships among associations when persistent interference from neighboring targets coexists with random interference or clutter. In such cases, joint association events become necessary to properly account for these dependencies.

A more complex scenario may arise due to the inherent finite resolution capability of signal processing systems. While each measurement is typically assumed to originate from either a target or clutter, an additional possibility must be considered: the merging of detections from multiple targets. Specifically, measurement z_3 could potentially result from such merging, representing an unresolved measurement. This introduces a fourth origin hypothesis for a measurement lying within the intersection of two validation regions.

The discussion highlights the challenges associated with associating measurements to tracks. The complete problem involves associating measurements at each time step, updating the track's sufficient statistic, and propagating it to the subsequent time step.

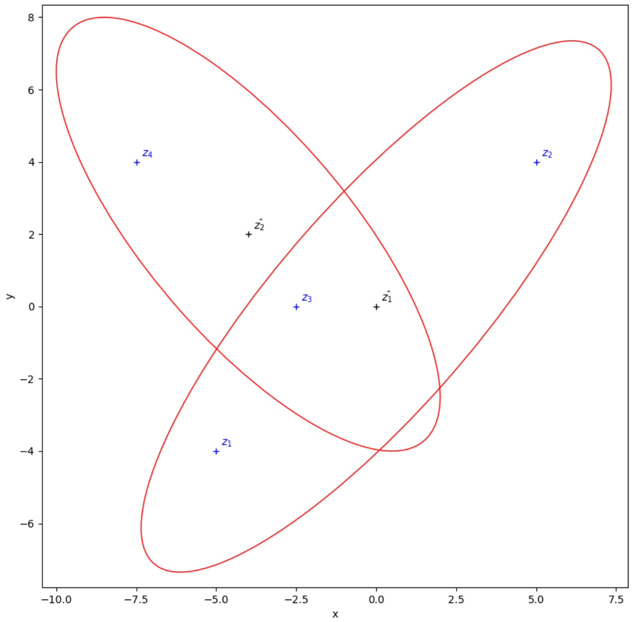


Figure 3.2 Several measurements z_i appeared in the validation region of one of targets $\hat{z_1}$ or $\hat{z_2}$. $\hat{z_1}$ and $\hat{z_2}$ are predicted measurements and none or any of the measurement $z_1 - z_3$ may have originated from the target $\hat{z_1}$ and none or any of the measurement $z_3 - z_4$ may have originated from the target $\hat{z_2}$.

3.3 Single target tracking

Single Target Tracking algorithms are specifically designed to track a single target in presence of clutter and other sources of interference. Unlike conventional Kalman filters, which is not inherently equipped to handle clutter. Single Target Tracking algorithms incorporate mechanisms to enhance target survivability in cluttered environments.

One prominent algorithm in the data association category of target tracking is the Probabilistic Data Association filter. The PDA filter addresses the challenge of associating measurements with the correct target while accounting for data association uncertainty. This uncertainty arises from the presence of clutter and the possibility of spurious measurements.

3.3.1 PDA filter

The Probabilistic Data Association algorithm computes the association probabilities for each validated measurement at the current time with respect to the target being tracked. This probabilistic or Bayesian information serves as a foundation for the Probabilistic Data Association Filter tracking algorithm, which effectively addresses the uncertainty regarding the origin of measurements. In scenarios where the state and measurement equations are linear, the resulting PDAF algorithm operates based on the Kalman Filter. However, if the state or measurement equations are non-linear, the PDAF algorithm is instead based on the Extended Kalman Filter. This adaptation allows the PDAF algorithm to accommodate non-linearity in the system dynamics or measurement processes, ensuring robust performance in diverse tracking scenarios.

The PDAF algorithm comes with many assumptions:

1. Only one target of interest is present, whose state $x \in R^{n_x}$ is assumed to evolve in time according to the equation

$$x_k = F_{k-1}x_{k-1} + w_{k-1}, \tag{3.1}$$

with the true measurement $z_k \in R^{n_z}$ given by

$$z_k = H_k x_k + v_k, \tag{3.2}$$

where w_{k-1} and v_k are zero mean mutually independent, white Gaussian noise variables with covariances Q_{k-1} and R_k, respectively.

2. The track has been initialized.

3. The past information through time $k-1$ about the target is summarized approximately by a suffifient statistic in the form of the Gaussian posterior

$$p(x_{k-1}|z_{k-1}) = \mathcal{N}(x_{k-1}; x_{k-1|k-1}, P_{k-1|k-1})). \tag{3.3}$$

4. At each time a measurement validation region is set up around the predicted measurement to select the candidate measurements for association to the target of interest.

5. If the target was detected and the corresponding measurement fell into the validation region, then, according to (3.2), at most one of the validated measurements can be target originated.

6. The remaining measurements are assumed to be due to false alarms or clutter and are modeled as independent and identically distributed with uniform spatial distribution, and the number of false alarms or clutter points obeys either a Poisson distribution, that is, a spatial Poisson process, with known spatial density λ, or a diffuse prior.

7. The target detections occur independently over time with known probability p_D.

All these assumptions make a filter for state estimation, that is almost as simple as Kalman filter, but more effective in the presence of clutter.

As well as the Kalman Filter, the PDAF also consists of prediction and update step. Before the update step, two additional steps occur – measurement validation and data association.

3.3.1.1 PDAF predict step

The prediction step of PDA filter from time $k-1$ to k is as in the standard KF,

$$x_{k|k-1} = F_{k-1}x_{k-1|k-1}, \tag{3.4}$$

$$\eta_{k|k-1} = H_k x_{k|k-1}, \tag{3.5}$$

$$P_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^T + Q_{k-1}, \tag{3.6}$$

where $P_{k-1|k-1}$ is from the equation (3.3). The innovation covariance matrix S_k is computed as

$$S_k = H_k P_{k|k-1} H_k^T + R_k. \tag{3.7}$$

3.3.1.2 PDAF measurement validation step

As discussed in Section 3.1.1, measurements considered to be originated from a given target should fall into validation region. This elliptical region is formulated as

$$\nu(k,\gamma) = \bigcup_{z \in Z_k} [(z - \hat{z}_{k|k-1})^T S_k^{-1} (z - \hat{z}_{k|k-1})) \leq \gamma], \tag{3.8}$$

where Z_k are all measurements at time k, γ is the gate threshold corresponding to the gate probability p_G, which is the probability, that the region contains the correct measurement if detected and S_k given in (3.7) is the covariance of the innovation.

3.3.1.3 PDAF data association step

The clutter in target tracking is assumed to be Poisson model with spatial density λ. In PDAF it yields to the association probability for $z_{i,k}$ being the correct measurement as

$$\beta_{i,k} = \begin{cases} \frac{\mathcal{L}_{i,k}}{1 - p_D p_G + \sum_{j=1}^{m_k} \mathcal{L}_{j,k}}, & i = 1, \ldots, m_k, \\ \frac{1 - p_D p_G}{1 - p_D P_G + \sum_{j=1}^{m_k} \mathcal{L}_{j,k}}, & i = 0, \end{cases} \tag{3.9}$$

where $i = 0$ means none is correct, p_D is the detection probability, p_G is the gate probability analogous to (3.8), and

$$\mathcal{L}_{i,k} = \frac{\mathcal{N}(z_{i,k}; \hat{z}_{k|k-1}, S_k) p_D}{\lambda} \tag{3.10}$$

is the likelihood ratio of the measurement $z_{i,k}$ from the validation region. The parameter λ (the density of the spatial Poisson clutter process) in the denominator of (3.10) models the clutter density of the uniform pdf of the location of false measurement. The Probabilistic Data Association algorithm produces association probabilities that are influenced by the position of the respective innovation on the Gaussian exponential of the likelihood ratio (3.10), in comparison to other innovations.

3.3.1.4 PDAF update step

The update step equation of update step is formulated as

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \nu_k, \tag{3.11}$$

where the combined innovation is

$$\nu_k = \sum_{i=1}^{m_k} \beta_{i,k} \nu_{i,k}, \tag{3.12}$$

and the Kalman gain is the same as in (2.50)

$$K_k = P_{k|k-1} H_k^T S_k^{-1}. \tag{3.13}$$

The covarince matrix associated with the update state (3.11) is expressed as

$$P_{k|k} = \beta_{0,k} P_{k|k-1} + (1 - \beta_{0,k}) P_{k|k}^c + \tilde{P}_k, \tag{3.14}$$

where the covariance of the updated state with the corrected measurement is

$$P^c_{k|k} = P_{k|k-1} - K_k S_k K_k^T, \tag{3.15}$$

and of the innovation term spreads according to

$$\tilde{P}_k = K_k(\sum_{i=1}^{m_k} \beta_{i.k} \nu_{i,k} \nu_{i,k}^T - \nu_{i,k} \nu_{i,k}^T) K_k^T. \tag{3.16}$$

With probability $\beta_{0,k}$ none of the measurement is correct. In such case, there is no update step of the state estimation. The prediction covariance $P_{k|k-1}$ in (3.14) has weight $\beta_{0,k}$. There is probability $1-\beta_{0,k}$ that correct measurement appears and the update covariance $P^c_{k|k}$ has weight $1-\beta_{0,k}$. However, there is no certainty, which of the validated measurement z_k is correct, so the positive semidefinite covariance matrix increases. This matrix is the result of the measurement origin uncertainty. Note, that in (3.16) the dependence of the estimation is nonlinear. The PDAF is nonlinear estimator, while the estimate update in (3.11) appears linear, but the association probabilities $\beta_{i,k}$ depend on the innovation in (3.9).

3.4 Multi target tracking

In the field of multi-target tracking, there are many different algorithms and approaches. One of the classes is the particle filters approach ([3], [23]–[25]). These filters are able to handle nonlinear motion, but the computational demands are really high. The second class is bayesian filtering, which is usually more effective when it comes to computational demands. Moreover there are two different approaches – data association filters and random finite sets filters . The example of data association, which we talked about in section 3.1, is the PDA filter (section 3.3.1) or the integrated probabilistic data association filter. The updated version of these filters, that can handle more targets, especially when two or more targets' validation region is in the same neighbourhood are the joint probabilistic data association filter and the joint integrated probabilistic data association filter. The other family is filters based on random finite sets. Random finite sets statistics are explained in the following section.

3.4.1 RFS statistics

Multi-target tracking presents challenges in estimating the states of multiple dynamic objects over time, complicated by varying target counts, cluttered sensor measurements, and data association ambiguity. Traditional tracking frameworks rely on explicit associations between measurements and targets, leading to computational complexities, particularly in scenarios with high target densities and clutter rates.

Recent advancements introduce Random Finite Set theory as an alternative approach to multi-target tracking. RFS theory represents sets with random cardinalities and values, offering a flexible framework for modeling uncertain multi-target scenarios. By treating both the multitarget state and sensor measurements as RFS, tracking algorithms can capture uncertainties effectively.

In scenario with more targets, i.e., multitarget scenario, let $M(k-1)$ be the number of targets at time k. At time $k-1$ the target states are $x_{k-1,1}, \dots, x_{k-1,M(k-1)} \in \mathcal{X}$. At the following time step k, some of these targets may disappear, some may evolve to their new states and also some new targets may be borned. As a result we have $M(k)$ targets with states $x_{k,1}, \dots, x_{k,M(k)} \in \mathcal{X}$. The RFS model formulation ensures, that the order in which the states are listed has no significance. Also, at time k, $N(k)$ measurements $z_{k,1}, \dots, z_{k,N(k)} \in \mathcal{Z}$ are received by the sensors, but the origins of these measurements are not known and the RFS model again ensures, that the order in which the measurements came, has no significance. Some of these measurements are generated by the targets, some are only false measurements, i.e., clutter.

The goal of multiple-target tracking is to collectively estimate both the quantity of targets and their respective states using measurements of uncertain origin. Even under ideal circumstances where the sensor detects all targets without clutter, single-target filtering techniques are inadequate due to the absence of information regarding the origin of each observation.

Given the absence of inherent ordering within the sets of target states and measurements at a specific time, they can be naturally depicted as finite sets, denoted as

$$X_k = \{x_{k,1}, \ldots, x_{k,M(k)}\} \in \mathcal{F}(\mathcal{X}) \tag{3.17}$$

$$Z_k = \{z_{k,1}, \ldots, z_{k,N(k)}\} \in \mathcal{F}(\mathcal{Z}), \tag{3.18}$$

where $\mathcal{F}(\mathcal{X})$ and $\mathcal{F}(\mathcal{Z})$ are collections of all finite subsets of $\mathcal{X}$ and $\mathcal{Z}$, respectively. In the random finite set approach, we consider the sets of targets and measurements, denoted as X_k and Z_k respectively, as the state and observation. This perspective allows us to frame the multi-target tracking problem as a filtering task with a state space $\mathcal{F}(\mathcal{X})$ and an observation space $\mathcal{F}(\mathcal{Z})$.

In a scenario with a single target, uncertainty is typically represented using random vectors to model the state x_k and the measurement z_k. Similarly, in a multi-target scenario, we represent uncertainty by using random finite sets to model the multie-target state X_k and the multi-target measurement Z_k. An RFS X is essentially a random variable that takes on finite sets of values, described by a discrete probability distribution and a set of joint probability densities. The distribution specifies the number of elements in X, while the densities describe the distribution of these elements.

We now delve into an RFS-based model for the evolution of the multi-target state, taking into account target motion, birth and death. Given a multi-target state X_{k-1} at time $k-1$, each individual target $x_{k-1} \in X_{k-1}$ either remains present at time k with probability $p_{S,k}(x_{k-1})$ or disappears with probability $1 - p_{S,k}(x_{k-1})$. If the target continues to exist, the probability density of transitioning from state x_{k-1} to x_k is denoted by $f_{k|k-1}(x_k|x_{k-1})$. Consequently, for each state $x_{k-1} \in X_{k-1}$ at time $k-1$, its behavior in the subsequent time step is modeled as the RFS

$$S_{k|k-1}(x_{k-1}), \tag{3.19}$$

that can represent two scenarios: either the survival of a target, indicated by the set $\{x_k\}$, or the absence of a target, denoted by $\emptyset$, implying the target's demise.

The occurrence of a new target at time k can originate from two possibilities: spontaneous births, which are independent of any existing target, or the spawning from a target present at time $k-1$.

Given a multi-target state X_{k-1} at time $k-1$, the state X_k at time k is formulated as the union of surviving targets, newly spawned targets, and spontaneously borned targets

$$X_k = \left[\bigcup_{\zeta \in X_{k-1}} S_{k|k-1}(\zeta)\right] \cup \left[\bigcup_{\zeta \in X_{k-1}} B_{k|k-1}(\zeta)\right] \cup \Gamma_k, \tag{3.20}$$

where

- Γ_k is a RFS of spontaneous births at time k,
- $B_{k|k-1}(\zeta)$ is a RFS of targets spawned at time k from a target with previous state ζ.

It is also assumed that RFSs in (3.20) are independent of each other, but the forms of Γ_k and $B_{k|k-1}(\cdot)$ are problem dependent.

The measurement sets Z_k are also random finite sets including detection uncertainty and clutter. A target $x_k \in X_k$ can be either detected with probability $p_{D,k}(x_k)$ or misdetected with probability $1 - p_{D,k}(x_k)$. Moreover, at time k, each state x_k produces an RFS

$$\Theta_k(x_k) \tag{3.21}$$

that is z_k in a case when the target is detected or $\emptyset$ otherwise. Furthermore, the sensor not only receives measurements originated from targets, but also a set K_k, which are false measurements (clutter). Thus, at time k, the multi-target measurement set Z_k observed by the sensor is also formulated as a union of measurements originated from targets and clutter

$$Z_k = \left[\bigcup_{x \in X_k} \Theta_k(x) \right] \cup \mathcal{K}_k. \tag{3.22}$$

The RFSs are assumed to be independent of each other and the actual form of K_k is problem dependent.

Finite Set Statistics (FISST) is fundamental within the RFS framework, providing a systematic way to apply RFS theory to multi-target tracking. FISST extends Bayesian filtering techniques to multi-target scenarios, enabling rigorous estimation of multi-target states in the presence of clutter and data association uncertainties. Datailed descriptions, informations and equations can be found in [26]. In this book, for sake of interest , we can find out that for a function $f = \partial F_{(\cdot)}(T)$ the set integral over a subset $S \subset E$ is defined as

$$\int_S f(X)\partial X \equiv \sum_{i=0}^{\infty} \frac{1}{i!} \int_{S^i} f(x_1, \ldots, x_i)\lambda_K^i(dx_1 \ldots dx_i), \tag{3.23}$$

where $\lambda_K(S)$ denote the hyper-volume of S in units of K, which is usually in MTT modeled as RFS Poisson process with uniform rate of K^{-1} with intensity $\lambda = \lambda_K/K$, i.e., [27]

$$\mu(\mathcal{T}) = \sum_{i=0}^{\infty} \frac{\lambda^i(\mathcal{T} \cap E^i)}{i!}, \tag{3.24}$$

where E and $\mathcal{T}$ are closed and bounded sebsets of R^n.

3.4.2 PHD filter

The PHD filter serves as an approximation devised to tackle the computational complexity inherent in the multi-target Bayes filter. Instead of tracking the complete posterior density of multi-target states over time, the PHD filter focuses on propagating the posterior intensity, which represents the first-order statistical moment of the posterior multi-target state [5]. This operational strategy draws parallels with the constant gain Kalman filter, which tracks the first moment (mean) of the single-target state [18].

For a Random finite set X on $\mathcal{X}$ with probability distribution P, its first-order moment is a non-negative function ν on $\mathcal{X}$, known as the intensity. This function satisfies the property that for each region $S \subseteq \mathcal{X}$ [28]

$$\int |X \cap S| P(dX) = \int_S \nu(x)dx \tag{3.25}$$

In essence, the integral of ν over any region S yields the expected number of elements of X within S. Consequently, the total mass $\hat{N} = \int \nu(x)dx$ represents the expected number of elements of X. The local maxima of the intensity ν correspond to points in $\mathcal{X}$ with the highest local concentration of expected number of elements, which can then be utilized to generate estimates for the elements of X. The simplest approach involves rounding $\hat{N}$ and selecting the resulting number of highest peaks from the intensity. In the tracking domain, this intensity is also referred to as the probability hypothesis density.

An important subclass of RFSs, known as Poisson RFSs, is characterized entirely by their intensities. An RFS X is considered Poisson if its cardinality distribution $Pr(|X| = n)$ follows a Poisson distribution with mean $\hat{N}$, and for any finite cardinality, the elements $x \in X$ are independently and identically distributed according to the probability density $\nu(\cdot)/N$. In the context of the multi-target tracking problem, it is common practice to model the clutter RFS [denoted as $\mathcal{K}_k$ in (3.22)] and the birth RFSs [Γ_k and $B_{k|k-1}(x_{k-1})$ in (3.20)] as Poisson RFSs.

To formulate PHD filter, let first denote multi-target state and measurement models with

$\gamma_k(\cdot)$	intensity of the births RFS Γ_k at time k;	(3.26)
$\beta_{k\|k-1}(\cdot\|\zeta)$	intensity of the RFS $B_{k\|k-1}(\zeta)$ spawned at time k by a target with previous state ζ;	(3.27)
$p_{S,k}(\zeta)$	probability that a target still exists at time k given that its previous state is ζ;	(3.28)
$p_{D,k}(x)$	probability of detection given a state x at time k;	(3.29)
$\kappa_k(\cdot)$	intensity of clutter RFS $\mathcal{K}_k$ at time k.	(3.30)

PHD filter naturally comes with certain assumptions.

1. Each target evolves and generates observations independently of each other.
2. Clutter is Poisson and independent of target-originated measurements.
3. The predicted multi-target RFS governed by $p_{k|k-1}$ is Poisson.

Assumptions 1 and 2 are very common in several tracking applications. The assumption 3 is acceptable approximation in situations with small correlation between targets and is satisfied when there is no spawning and RFSs X_{k-1} and Γ_k are Poisson.

3.4.2.1 PHD filter recursion

Let assume a target state x described by an intensity function $\nu(x)$. At time k, the prediction of the prior intensity $\nu_{k-1}(x)$ is given by

$$\begin{aligned}\nu_{k|k-1}(x) = &\int p_{S,k}(\zeta)\phi_{k|k-1}(x|\zeta)\nu_{k-1}(\zeta)d\zeta \\ &+ \int \beta_{k|k-1}(x|\zeta)\nu_{k-1}(\zeta)d\zeta + \nu_{\gamma,k}(x),\end{aligned} \tag{3.31}$$

where $p_{S,k}(\cdot)$ is the probability of target survival, $\phi_{k|k-1}(\cdot|\cdot)$ is the target state transition density, and $\nu_{\gamma,k}(\cdot)$ denotes the prior PHD of the target's birth at time k. The predicted intensity $\nu_{k|k-1}$ is then updated by the measurement set Z_k given by sensors at time k according to the Bayesian update

$$\nu_k(x) = [1 - p_{D,k}(x)]\nu_{k|k-1}(x) \tag{3.32}$$

$$+ \sum_{z \in Z_k} \frac{p_{D,k}(x)g_k(z|x)\nu_{k|k-1}(x)}{\kappa_k(z) + \int p_{D,k}(\zeta)g_k(z|\zeta)\nu_{k|k-1}(\zeta)d\zeta}, \tag{3.33}$$

where $g_k(\cdot|\cdot)$ is the likelihood function, $p_{D,k}(\cdot)$ is the probability of detection, and $\kappa_k(\cdot)$ is the clutter density.

Equations (3.31) and (3.33) show that the PHD filter avoids the combinatorial ineffectivity awakening from the unknown association of measurements with correct targets. The posterior intensity is a function of a single-target state space $\mathcal{X}$, thus the recursion is more computationally effective than (2.34) and (2.35) that in the case of the PHD filter operates on $\mathcal{F}(\mathcal{X})$. But in general it is not possible to come up with a closed-form solution in the PHD recursion.

3.4.2.2 GM-PHD for linear models

In this section we delve into details of multi-target models, i. e., linear Gaussian multi-target models. With PHD recursions (3.31) and (3.33) it is achievable to get an efficient multi-taget closed form solution for multi-target algorithms. This solution proposed by Vo and Ma in [18] requires additional assumptions to the already proposed ones. Not only standard linear Gaussian model for each targets are assumed, but also the linear Gausssian multi-target model includes assumptions on the death, birth and detections.

4. Each target follows a linear Gaussian dynamical model and the sensor has a linear Gaussian measurement model, i.e.,

$$f_{k|k-1}(x|\zeta) = \mathcal{N}(x; F_{k-1}\zeta, Q_{k-1}) \tag{3.34}$$

$$g_k(z|x) = \mathcal{N}(z; H_k x, R_k), \tag{3.35}$$

   where $\mathcal{N}(\cdot; m, P)$ is a Gaussian density with mean m and covariance P, F_{k-1} is the state transition matrix, Q_{k-1} is the process noise covariance, H_k is the observation matrix and R_k is the observation noise covariance.

5. The survival and detection probabilities are state independent, i. e.,

$$p_{S,k}(x) = p_{S,k} \tag{3.36}$$

$$p_{D,k}(x) = p_{D,k}. \tag{3.37}$$

6. The intesities of the birth and spawn RFSs are Gaussian mixtures of the form

$$\gamma_k(x) = \sum_{i=1}^{J_{\gamma,k}} w_{\gamma,k}^{(i)} \mathcal{N}(x; m_{\gamma.k}^{(i)}, P_{\gamma,k}^{(i)}) \tag{3.38}$$

$$\beta_{k|k-1}(x|\zeta) = \sum_{j=1}^{J} {}_{\beta,k} w_{\beta,k}^{(j)} \mathcal{N}(x; F_{\beta,k-1}^{(} j)\zeta + d_{\beta,k-1}^{(j)}, Q_{\beta,k-1}^{(j)}), \tag{3.39}$$

where $J_{\gamma,k}, w_{\gamma,k}^{(i)}, m_{\gamma,k}^{(i)}, P_{\gamma,k}^{(i)}, i = 1, \ldots, J_{\gamma,k}$ are given model parameters that actuate the shape of the birth intensity and $J_{\beta,k}, w_{\beta,k}^{(j)}, F_{\beta,k-1}^{(j)}, d_{\beta,k-1}^{(j)}$ and $Q_{\beta,k-1}^{(j)}, j = 1, \ldots, J_{\beta,k}$ determine the shape of the spawning intensity of a target with previous state ζ.

Assumptions 4 and 5 are very common and used in various tracking algorithms [29], but the closed-form solution allows for state- and time-dependent $p_{S,k}$ and $p_{D,k}$. This convenience is used in the practical part of this thesis.

The means $m_{\gamma,k}^{(i)}$ is assumption 6 are the peaks of the spontaneous birth intensity in (3.38). These are points with highest concentrations of expected number of spontaneous births and represent place like airports or other places with highest probability targets appear. $P_{\gamma,k}^{(i)}$ is the covariance matrix that regulate the spread of the birth intensity. The weight $w_{\gamma,k}^{(i)}$ gives the expected number number of new targets originating from $m_{\gamma,k}^{(i)}$. Similary in (3.39) with addition, that the peak $F_{\beta,k-1}^{(j)}\zeta + d_{\beta,k-1}^{(j)}$ is an affine function of ζ. This spawned target can be pictured as a new target spawned in the neighborhood of another target with previous state ζ. This can be, for example, a smaller ship detached from its mother ship.

3.4.2.3 GM-PHD recursion for linear models

In this section we delve into Gaussian mixture probability hypothesis density filter recursion for the linear Gaussian multi-target model and its closed-form solution to the (3.31) and (3.33). With assumptions 4-6 suppose that the posterior intensity at time $k-1$ is a Gaussian mixture of the form

$$\nu_{k-1}(x) = \sum_{i=1}^{J_{k-1}} w_{k-1}^{(i)} \mathcal{N}(x; , m_{k-1}^{(i)}, P_{k-1}^{(i)}). \tag{3.40}$$

Then, the predicted intensity for time k is also a Gaussian mixture and is given by

$$\nu_{k|k-1}(x) = \nu_{S,k|k-1}(x) + \nu_{\beta,k|k-1}(x) + \gamma_k(x), \tag{3.41}$$

where $\gamma_k(x)$ is given in (3.38). This yields

$$\nu_{S,k|k-1}(x) = p_{S,k} \sum_{j=1}^{J_{k-1}} w_{k-1}^{(j)} \mathcal{N}(x; m_{S,k|k-1}^{(j)}, P_{S,k|k-1}^{(j)}), \tag{3.42}$$

$$m_{S,k|k-1}^{(j)} = F_{k-1} m_{k-1}^{(j)}, \tag{3.43}$$

$$P_{S,k|k-1}^{(j)} = Q_{k-1} + F_{k-1} P_{k-1}^{(j)} F_{k-1}^T \tag{3.44}$$

$$\nu_{\beta,k|k-1}(x) = \sum_{j=1}^{J_{k-1}} \sum_{l=1}^{J_{\beta,k}} w_{k-1}^{(j)} w_{\beta,k}^{(l)} \mathcal{N}(x; m_{\beta,k|k-1}^{(j,l)}, P_{\beta,k|k-1}^{(j,l)}) \tag{3.45}$$

$$m_{\beta,k|k-1}^{(j,l)} = F_{\beta,k-1}^{(l)} m_{k-1}^{(j)} + d_{\beta,k-1}^{(l)} \tag{3.46}$$

$$P_{\beta,k|k-1}^{(j,l)} = Q_{\beta,k-1}^{(l)} + F_{\beta,k-1}^{(l)} P_{\beta,k-1}^{(l)} (F_{\beta,k-1}^{(l)})^T. \tag{3.47}$$

Thus the predicted intensity at the time k is a Gaussian mixture

$$\nu_{k|k-1}(x) = \sum_{i=1}^{J_{k|k-1}} w_{k|k-1}^{(i)} \mathcal{N}(x; m_{k|k-1}^{(i)}, P_{k|k-1}^{(i)}), \tag{3.48}$$

and the posterior intensity at time k is also Gaussian mixture,

$$\nu_k(x) = (1 - p_{D,k}) \nu_{k|k-1}(x) + \sum_{z \in Z_k} \nu_{D,k}(x; z), \tag{3.49}$$

where

$$\nu_{D,k}(x; z) = \sum_{j=1}^{J_{k|k-1}} w_k^{(j)}(z) \mathcal{N}(x; m_{k|k}^{(j)}(z), P_{k|k}^{(j)}), \tag{3.50}$$

$$w_k^{(j)}(z) = \frac{p_{D,k}^{(j)}(x; z) w_{k|k-1}^{(j)} q_k^{(j)}(z)}{\kappa_k(z) + p_{D,k}^{(j)}(x; z) \sum_{l=1}^{J_{k|k-1}} w_{k|k-1}^{(l)} q_k^{(l)}(z)}, \tag{3.51}$$

$$m_{k|k}^{(j)}(z) = m_{k|k-1}^{(j)} + K_k^{(j)}(z - H_k m_{k|k-1}^{(j)}), \tag{3.52}$$

$$P_{k|k}^{(j)}(z) = [I - K_k^{(j)} H_k] P_{k|k-1}^{(j)}, \tag{3.53}$$

$$K_k^{(j)}(z) = P_k^{(j)} H_k^T (H_k P_{k|k-1}^{(j)} H_k^T + R_k)^{-1}. \tag{3.54}$$

Formulations (3.40), (3.41) and (3.48), (3.49) are the result of applying standard Gaussian properties, i. e., given F, d, Q, m and P of appropriate dimensions and that Q and P are positive definite matrices

$$\int \mathcal{N}(x; F\zeta + d, Q)\mathcal{N}(\zeta; m, P)d\zeta = \mathcal{N}(x; Fm + d, Q + FPF^T) \tag{3.55}$$

and given H, R, m and P of appropriate dimensions assuming R and P are positive definite matrices

$$\mathcal{N}(z; Hx, R)\mathcal{N}(x; m, P) = q(z)\mathcal{N}(x; \tilde{m}, \tilde{P}), \tag{3.56}$$

where

$$q(z) = \mathcal{N}(z; Hm, R + HPH^T) \tag{3.57}$$

$$\tilde{m} = m + K(z - Hm) \tag{3.58}$$

$$\tilde{P} = (I - KH)P \tag{3.59}$$

$$K = PH^T(HPH^T + R)^{-1}. \tag{3.60}$$

Equations (3.34), (3.36), and (3.38) - (3.40) are inserted into the PHD prediction equation (3.31), and then replacing integrals of the type (3.55) with suitable Gaussian functions. Similarly, formulations (3.48), (3.49) are established by substituting equations (3.35), (3.37), and (3.48) into the PHD update equation (3.33), and subsequently replacing integrals resembling form (3.55) and products of Gaussians resembling form (3.56) with appropriate Gaussian functions.

If we comply prior intensity ν_0 as a Gaussian mixture, then all following predicted intensities $\nu_{k|k-1}$ and posterior intensities ν_k are also Gaussian mixtures and the recursion holds. Formulations (3.40) and (3.41) give a closed-form guidance for computing the means, covariances and weights of ν_k from $\nu_{k|k-1}$ after new set of measurements is available.

The intensity $\nu_{k|k-1}$ in (3.41) includes three intensities $\nu_{S,k|k-1}, \nu_{,k|k-1}$ and γ_k. These intensities represent existing targets, spawned targets and spontaneous births. The updated intensity ν_k consists a misdetection case $(1 - p_{D,k})\nu_{k|k-1}$ and $|Z_k|$ detection cases $\nu_{D,k}(;z)$ for every $z \in Z_k$. The GM-PHD recursion prediction step thus follows Kalman prediction step and and the GM-PHD update step follows Kalman update step.

By summing the weights of $\nu_{k|k-1}$ and ν_k we can get the expected number of targets $\tilde{N}_{k|k-1}$ and $\tilde{N}_k$, respectively,

$$\tilde{N}_{k|k-1} = \tilde{N}_{k-1}\left(p_{S,k} + \sum_{j=1}^{J_{\beta,k}} w_{\beta,k}^{(j)}\right) + \sum_{j=1}^{J_{\gamma,k}} w_{\gamma,k}^{(j)} \tag{3.61}$$

$$\tilde{N}_k = \tilde{N}_{k|k-1}(1 - p_{D,k}) + \sum_{z\in Z_k}\sum_{j=1}^{J_{k|k-1}} w_k^{(j)}(z), \tag{3.62}$$

where $\tilde{N}_{k|k-1}$ is expected number of targets after prediction step, which is the sum of surviving, spawning and borned targets. Similary, $\tilde{N}_k$ is the expected number of targets after update step, which is the sum of updated number of targets plus the mean number of targets that are not detected.

The GM-PHD filter pseudo algorithm is provided in algorithm 2.

3.4.2.4 Pruning and merging in GM-PHD filter

Due to the propogation of the Gaussian mixture in time, the PHD filter suffers from computation issues resulting from exponential increase of Gaussian components

$$(J_{k-1}(1 + J_{\beta,k}) + J_{\gamma,k})(1 + |Z_k|) = \mathcal{O}(J_{k-1}|Z_k|), \tag{3.63}$$

where J_{k-1} is the number of Gaussian components of intensity ν_{k-1}.

To resolve this issue, many techniques can be applied. The simplest one is to remove components with low weight to the next timestep. This can be done be predefining a threshold, under which the target falls, is removed.

The next possibility is the keep a certain number of targets and remove the ones with lowest weight. However, some of the Gausssian components can be so close together, that they can be merged into a single component. Such procedure is a good approximation with appropriate merging threshold. Pseudo algorithm for this procedure is shown in algorithm 3.

After computing the posterior intensity ν_k, the subsequent objective is to derive estimates for the multiple-target states. In the context of the Gaussian mixture representation of ν_k, extracting estimates for the multiple-target states becomes relatively straightforward, as the means of the individual Gaussian components typically correspond to the local maxima of ν_k, given that they are adequately separated. However, it's essential to note that closely spaced Gaussian components may have been merged after pruning.

Since the magnitude of each peak is influenced by both its weight and covariance, merely selecting the $\tilde{N}_k$ highest peaks of ν_k may yield state estimates associated with Gaussians having weak weights. This scenario is suboptimal, as the expected number of targets attributed to these peaks might be small despite their peak magnitudes being substantial. A more effective approach involves choosing the means of the Gaussians with weights exceeding a certain threshold, such as 0.5. This method ensures a more robust selection of state estimates.

The outlined procedure for state estimation in the Gaussian mixture PHD filter is summarized in algorithm 4.

Algorithm 2 Pseudo algorithm for the GM-PHD filter

Input: $\{w_{k-1}^{(i)}, m_{k-1}^{(i)}, P_{k-1}^{(i)}\}_{i=1}^{J_{k-1}}$, and the measurement set Z_k.

Function **Prediction step for birth targets:**
  for $j = 1, \ldots, J_{\gamma,k}$ **do**
    $i := i + 1,$
    $w_{k|k-1}^{(i)} = w_{\gamma,k}^{(i)}, \quad m_{k|k-1}^{(i)} = m_{\gamma,k}^{(i)}, \quad P_{k|k-1}^{(i)} = P_{\gamma,k}^{(i)},$
  end for
  for $j = 1, \ldots, J_{\beta,k}$ **do**
    for $l = 1, \ldots, J_{k-1}$ **do**
      $i := i + 1,$
      $w_{k|k-1}^{(i)} = w_{k-1}^{(l)} w_{\beta,k}^{(j)},$
      $m_{k|k-1}^{(i)} = d_{\beta,k-1}^{(j)} + F_{\beta,k-1}^{(j)} m_{k-1}^{(l)},$
      $P_{k|k-1}^{(i)} = Q_{\beta|k-1}^{(j)} + F_{\beta,k-1}^{(j)} P_{k-1}^{(l)} (F_{\beta,k-1}^{(j)})^T.$
    end for
  end for
end *Function*

Function **Prediction step for existing targets:**
  for $j = 1, \ldots, J_{k-1}$ **do**
    $i := i + 1,$
    $w_{k|k-1}^{(i)} = p_{S,k} w_{k-1}^{(j)},$
    $m_{k|k-1}^{(i)} = F_{k-1} m_{k-1}^{(j)},$
    $P_{k|k-1}^{(i)} = Q_{k-1} + F_{k-1} P_{k-1}^{(j)} F_{k-1}^T,$
  end for
  $J_{k|k-1} = i$
end *Function*

Function **Computation of PHD update components:**
  for $j = 1, \ldots, J_{k|k-1}$ **do**
    $\eta_{k|k-1}^{(j)} = H_k m_{k|k-1}^{(j)}, \quad S_k^{(j)} = R_k + H_k P_{k|k-1}^{(j)} H_k^T,$
    $K_k^{(j)} = P_{k|k-1}^{(j)} H_k^T [S_k^{(j)}]^{-1}, \quad P_{k|k}^{(j)} = [I - K_k^{(j)} H_k] P_{k|k-1}^{(j)}.$
  end for
end *Function*

Function **PHD update step:**
  for $j = 1, \ldots, J_{k|k-1}$ **do** ▷ Misdetection
    $w_k^{(j)} = (1 - p_{D,k}) w_{k|k-1}^{(j)}, \quad m_k^{(j)} = m_{k|k-1}^{(j)}, \quad P_k^{(j)} = P_{k|k-1}^{(j)}$
  end for
  $l := 0$
  for all $z \in Z_k$ **do** ▷ Detection
    $l := l + 1,$
    for $j = 1, \ldots, J_{k|k-1}$ **do**
      $w_k^{(lJ_{k|k-1}+j)} = p_{D,k} w_{k|k-1}^{(j)} \mathcal{N}(z; \eta_{k|k-1}^{(j)}, S_k^{(j)}),$
      $m_k^{(lJ_{k|k-1}+j)} = m_{k|k-1}^{(j)} + K_k^{(j)} (z - \eta_{k|k-1}^{(j)}),$
      $P_k^{(lJ_{k|k-1}+j)} = P_{k|k}^{(j)},$
    end for
    $w_k^{(lJ_{k|k-1}+j)} := \frac{w_k^{(lJ_{k|k-1}+j)}}{\kappa_k(z) + \sum_{i=1}^{J_{k|k-1}} w_k^{(lJ_{k|k-1}+i)}}$, for $j = 1, \ldots, J_{k|k-1},$
  end for
  $J_k = lJ_{k|k-1} 9 J_{k|k-1}.$
end *Function*

Output: $\{w_k^{(i)}, m_k^{(i)}, P_k^{(i)}\}_{i=1}^{J_k}.$

Algorithm 3 Pseudo algorithm for pruning in the GM-PHD filter

Input: $\{w_k^{(i)}, m_k^{(i)}, P_k^{(i)}\}_{i=1}^{J_k}$, a truncation threshold T, a merging threshold U and a maximum number of allowed Gaussian terms J_{max}. Set $l = 0$, and $I = \{i = 1, \ldots, J_k | w_k^{(i)} > T\}$.

Function **Merging of targets**
  while $I \neq \emptyset$ **do**
    $l := l + 1$
    $j := \text{argmax}_{i \in I} w_k(i)$,
    $L := \left\{ i \in I | (m_k^{(i)} - m_k^{(j)})^T (P_k(i))^{-1} (m_k^{(i)} - m_k^{(j)}) \leq U \right\}$,
    $\tilde{w}_k^{(l)} = \sum_{i \in L} w_k^{(i)}$,
    $\tilde{m}_k^{(l)} = \frac{1}{\tilde{w}_k^{(l)}} \sum_{i \in L} w_k(i) x_k^{(i)}$,
    $\tilde{P}_k^{(l)} = \frac{1}{\tilde{w}_k^{(l)}} \sum_{i \in L} w_k^{(i)} (P_k^{(i)} + (m_k^{(l)} - m_k^{(i)})(m_k^{(l)} - m_k^{(i)})^T$,
    $I := I/L$.
  end while
end *Function*

If $l > J_{max}$ then replace $\{\tilde{w}_k^{(i)}, \tilde{m}_k^{(i)}, \tilde{P}_k^{(i)}\}_{i=1}^{l}$ by those of the $J_m ax$ Gaussians with largest weights.

Output: $\{\tilde{w}_k^{(i)}, \tilde{m}_k^{(i)}, \tilde{P}_k^{(i)}\}_{i=1}^{l}$ as pruned Gaussian components.

Algorithm 4 Pseudo algorithm for state extraction in the GM-PHD filter

Input: $\{w_k^{(i)}, m_k^{(i)}, P_k^{(i)}\}_{i=1}^{J_k}$.

Set $\hat{X}_k = \emptyset$
for $i = 1, \ldots, j_k$ **do**
  if $w_k^{(i)} > 0.5$, **then**
    for $j = 1, \ldots, round(w_k^{(i)})$ **do**
      update $\hat{X}_k := [\hat{X}_k, m_k^{(i)}]$
    end for
  end if
end for

Output: $\hat{X}_k$ as the multi-target state estimate.

Chapter 4

Object detection and segmentation

4.1 YOLO

4.2 Segment Anything

Chapter 5

Dynamic time and state varying detection probability

Chapter 6

Experiments

Appendix A

Nějaká příloha

Sem přijde to, co nepatří do hlavní části.

Bibliography

[1] Y. Bar-Shalom and X. Li, *Multitarget-multisensor Tracking: Principles and Techniques*. Yaakov Bar-Shalom, 1995, ISBN: 9780964831209.

[2] B. A., *Kalman Filter from the ground up*. Alex Becker, 2023, ISBN: 978-965-93120-1-6.

[3] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002. DOI: 10.1109/78.978374.

[4] B. E., *Fundamentals of Sensor Fusion*. Norwegian University of Science and Technology, 2020.

[5] R. Mahler, "Multitarget bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003. DOI: 10.1109/TAES.2003.1261119.

[6] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler, "Mot16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*, 2016.

[7] G. Hendeby and R. Karlsson, "Gaussian mixture phd filtering with variable probability of detection," in *17th International Conference on Information Fusion (FUSION)*, 2014, pp. 1–7.

[8] R. P. S. Mahler, B.-T. Vo, and B.-N. Vo, "Cphd filtering with unknown clutter rate and detection profile," *IEEE Transactions on Signal Processing*, vol. 59, no. 8, pp. 3497–3513, 2011. DOI: 10.1109/TSP.2011.2128316.

[9] B.-T. Vo, B.-N. Vo, R. Hoseinnezhad, and R. P. S. Mahler, "Robust multi-bernoulli filtering," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 3, pp. 399–409, 2013. DOI: 10.1109/JSTSP.2013.2252325.

[10] G. Li, L. Kong, W. Yi, and X. Li, "Robust poisson multi-bernoulli mixture filter with unknown detection probability," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 1, pp. 886–899, 2021. DOI: 10.1109/TVT.2020.3047107.

[11] C. Li, W. Wang, T. Kirubarajan, J. Sun, and P. Lei, "Phd and cphd filtering with unknown detection probability," *IEEE Transactions on Signal Processing*, vol. 66, no. 14, pp. 3784–3798, 2018. DOI: 10.1109/TSP.2018.2835398.

[12] J. Wei, F. Luo, J. Qi, and L. Ruan, "A modified bgm-phd filter with unknown detection probability," in *2023 6th International Conference on Information Communication and Signal Processing (ICICSP)*, 2023, pp. 492–496. DOI: 10.1109/ICICSP59554.2023.10390615.

[13] E. Hanusa and D. W. Krout, "Track state augmentation for estimation of probability of detection in multistatic sonar data," in *2013 Asilomar Conference on Signals, Systems and Computers*, 2013, pp. 1733–1737. DOI: 10.1109/ACSSC.2013.6810598.

[14] S. Horn, "Near real time estimation of surveillance gaps," in *Proceedings of the 16th International Conference on Information Fusion*, 2013, pp. 1871–1877.

[15] S. Wei, B. Zhang, and W. Yi, "Trajectory phd and cphd filters with unknown detection profile," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 8, pp. 8042–8058, 2022. DOI: 10.1109/TVT.2022.3174055.

[16] S. Yan, Y. Fu, W. Zhang, W. Yang, R. Yu, and F. Zhang, "Multi-target instance segmentation and tracking using yolov8 and bot-sort for video sar," in *2023 5th International Conference on Electronic Engineering and Informatics (EEI)*, 2023, pp. 506–510. DOI: 10.1109/EEI59236.2023.10212903.

[17] Á. F. García-Fernández, J. L. Williams, K. Granström, and L. Svensson, "Poisson multi-bernoulli mixture filter: Direct derivation and implementation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 4, pp. 1883–1901, 2018. DOI: 10.1109/TAES.2018.2805153.

[18] B.-N. Vo and W.-K. Ma, "The gaussian mixture probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, 2006. DOI: 10.1109/TSP.2006.881190.

[19] Y. Bulut, D. Vines-Cavanaugh, and D. Bernal, "Process and measurement noise estimation for kalman filtering," *Conference Proceedings of the Society for Experimental Mechanics Series*, vol. 3, May 2011. DOI: 10.1007/978-1-4419-9834-7_36.

[20] U. Hadar and H. messer, "High-order hidden markov models - estimation and implementation," in *2009 IEEE/SP 15th Workshop on Statistical Signal Processing*, 2009, pp. 249–252. DOI: 10.1109/SSP.2009.5278591.

[21] K. Dedecius and P. M. Djurić, "Sequential estimation and diffusion of information over networks: A bayesian approach with exponential family of distributions," *IEEE Transactions on Signal Processing*, vol. 65, no. 7, pp. 1795–1809, 2017. DOI: 10.1109/TSP.2016.2641380.

[22] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *IEEE Control Systems Magazine*, vol. 29, no. 6, pp. 82–100, 2009.

[23] Z. Khan, T. Balch, and F. Dellaert, "Mcmc-based particle filtering for tracking a variable number of interacting targets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1805–1819, 2005. DOI: 10.1109/TPAMI.2005.223.

[24] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, 2002. DOI: 10.1109/78.978396.

[25] A. Doucet, N. Gordon, and V. Krishnamurthy, "Particle filters for state estimation of jump markov linear systems," *IEEE Transactions on Signal Processing*, vol. 49, no. 3, pp. 613–624, 2001. DOI: 10.1109/78.905890.

[26] I. Goodman, R. Mahler, and H. Nguyen, *Mathematics of Data Fusion*, ser. Theory and Decision Library B. Springer Netherlands, 1997, ISBN: 9780792346746.

[27] B.-N. Vo, S. Singh, and A. Doucet, "Random finite sets and sequential monte carlo methods in multi-target tracking," in *2003 Proceedings of the International Conference on Radar (IEEE Cat. No.03EX695)*, 2003, pp. 486–491. DOI: 10.1109/RADAR.2003.1278790.

[28] D. Daley and D. Vere-Jones, *An introduction to the theory of point processes*. Springer, Nov. 2003.

[29] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*, ser. Mathematics in science and engineering. Academic Press, 1988, ISBN: 9780120797608.

Obsah příloh

```
 readme.txt ............ stručný popis obsahu média
 exe ................... adresář se spustitelnou formou implementace
 src
   impl ................ zdrojové kódy implementace
   thesis .............. zdrojová forma práce ve formátu LaTeX
 text .................. text práce
   thesis.pdf .......... text práce ve formátu PDF
```