The Kernel-SME Filter with False and Missing Measurements

Marcus Baum, Shishan Yang
Institute of Computer Science
University of Göttingen, Germany
Email: {marcus.baum, shishan.yang}@cs.uni-goettingen.de

Uwe D. Hanebeck
Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
Institute for Anthropomatics and Robotics
Karlsruhe Institute of Technology (KIT), Germany
Email: uwe.hanebeck@ieee.org

***Abstract*—The recently proposed Kernel-SME filter for multi-object tracking is a further development of the Symmetric Measurement Equation (SME) idea introduced by Kamen in the 1990s. The Kernel-SME constructs a symmetric, i.e., permutation invariant, measurement equation by transforming the measurements to a kernel mixture function. This transformation is scalable to a large number of objects and allows for deriving an efficient closed-form Gaussian filter based on the Kalman filter formulas. This work shows how the Kernel-SME approach can systematically incorporate false and missing measurements.**

***Keywords*—Multi-Object Tracking, Data Association, Symmetric Measurement Equation.**

I. Introduction

Multi-object tracking deals with the problem of successively estimating the number and states of multiple objects based on noisy unlabeled measurements [3], [15]. One of the main challenges in multi-object tracking is that the measurements are unlabeled, i.e., it is unknown which measurement belongs to which object. As the number of feasible association hypotheses explodes with the number of objects and measurements, sophisticated approximations are required in order to obtain efficient algorithms. In this context, many different multi-object tracking algorithms have been developed, e.g., the *Joint Probabilistic Data Association Filter (JPDAF)* [1] computes a Gaussian approximation of the posterior density of the objects states and the *Probability Hypothesis Density (PHD) filter* [14], [18] is based on Finite Set Statistics (FISST).

The Kernel-SME [4], [10] filter discussed in this paper is based on the key idea of the *Symmetric Measurement Equation (SME)* filter [8], [9], [12], [13], [17], [19], [20]. The SME filter avoids an explicit enumeration of data association hypotheses by constructing a pseudo-measurement with a symmetric transformation. A symmetric transformation is permutation invariant to the order of its arguments, so that the correct permutation, i.e., association, is not required. We argue that the original SME approach comes with three main problems: (i) existing SMEs suffer from strong nonlinearities, (ii) do not scale well with the number of measurements, and (iii) cannot systematically deal with false and missing measurements. We note that [11] introduces an approach for incorporating false and missing measurements in the original SME approach based on explicit enumeration of association hypotheses.

In our previous work about the Kernel-SME filter [4], [5], we addressed problem (i) and (ii) by proposing a novel SME that maps the measurements to a kernel mixture function. In this manner, a data-dependent symmetric transformation is constructed that scales well with the number of measurements. In this work, we address problem (iii): We show that clutter and missed detections can be systematically incorporated in the Kernel-SME framework, i.e., modeled within the symmetric transformation without the enumeration of association hypotheses.

In the following section, we describe the considered multi-object tracking problem. Section III introduces the novel symmetric measurement equation for false and missing measurements. Based on this measurement equation, we subsequently derive a Gaussian state estimator using the Kalman filter formulas in Section IV. After presenting numerical simulations that demonstrate the performance of the Kernel-SME filter in Section V, the paper is concluded in Section VI.

II. Problem Formulation

We treat the tracking of multiple objects based on noisy unlabeled measurements, i.e., the measurement-to-object association is unknown. Furthermore, we impose the following assumptions:

- A1 The number of objects N is known.
- A2 Each object gives rise to at most one measurement per scan/frame. The detection probability is denoted as p_d.
- A3 The number of false measurements is Poisson distributed with mean λ.

The d_x-dimensional object state vectors are denoted as $\underline{\boldsymbol{x}}_k^1, \ldots, \underline{\boldsymbol{x}}_k^N$, where k denotes the discrete time index.[1] The single object states are stacked in a *joint state vector* $\underline{\boldsymbol{x}}_k = \left[(\underline{\boldsymbol{x}}_k^1)^T, \ldots, (\underline{\boldsymbol{x}}_k^N)^T\right]^T \in \mathbb{R}^{d_x \cdot N}$.

A. Measurement Model

At each time k, a set of M_k measurements

$$Y_k = \{\underline{\boldsymbol{y}}_k^1, \ldots, \underline{\boldsymbol{y}}_k^{M_k}\} \subseteq \mathbb{R}^{d_y}$$

1 Vectors are underlined, e.g., $\underline{x}$, and random variables are printed in bold, e.g., $\boldsymbol{x}$ and $\underline{\boldsymbol{x}}$.

is available. The measurement set is related to the states according to

$$Y_k = C_k \cup \bigcup_{1\leq l\leq N} \Theta_k^l \tag{1}$$

where

- C_k consists of M_k^c clutter measurements that are uniformly distributed in the tracking region. The number of clutter measurements M_k^c is Poisson distributed with mean λ.
- Θ_k^l consists of the measurement from object l (if detected), i.e.,

$$\Theta_k^l = \begin{cases} \mathbf{H}_k^i \underline{\boldsymbol{x}}_k^l + \underline{\boldsymbol{v}}_k^l & \text{if detected} \\ \emptyset & \text{otherwise} \end{cases} \quad , \tag{2}$$

  where $\mathbf{H}_k^l$ is the measurement matrix and $\underline{\boldsymbol{v}}_k^l$ zero-mean noise with covariance matrix $\Sigma_{k,l}^v$.

B. System Model

The motion model of an individual object is specified by

$$\underline{\boldsymbol{x}}_{k+1}^l = \mathbf{A}_k^l \underline{\boldsymbol{x}}_k^l + \underline{\boldsymbol{w}}_k^l \quad , \tag{3}$$

where $\mathbf{A}_k^l$ is the system matrix and $\underline{\boldsymbol{w}}_k^l$ is additive white noise with covariance matrix $\Sigma_k^{w_l}$. All individual motion models (3) can be stacked into a single vector

$$\underbrace{\begin{bmatrix} \underline{\boldsymbol{x}}_{k+1}^1 \\ \vdots \\ \underline{\boldsymbol{x}}_{k+1}^N \end{bmatrix}}_{=\underline{\boldsymbol{x}}_{k+1}} = \underbrace{\begin{bmatrix} \mathbf{A}_k^1 & & \\ & \ddots & \\ & & \mathbf{A}_k^N \end{bmatrix}}_{:=\mathbf{A}_k} \cdot \underbrace{\begin{bmatrix} \underline{\boldsymbol{x}}_k^1 \\ \vdots \\ \underline{\boldsymbol{x}}_k^N \end{bmatrix}}_{=\underline{\boldsymbol{x}}_k} + \underbrace{\begin{bmatrix} \underline{\boldsymbol{w}}_k^1 \\ \vdots \\ \underline{\boldsymbol{w}}_k^N \end{bmatrix}}_{=\underline{\boldsymbol{w}}_k} \quad . \tag{4}$$

III. Kernel-SME

The measurement equation (1) relates the object states with the measurements. However, it relates two sets and it is not clear which element on the left-hand side is associated to which element on right-hand side, i.e., the association is not known. The basic idea of the Kernel-SME approach is to convert the *set-valued equation* (1) to an equivalent *vector-valued* equation using a symmetric transformation of the measurements. This transformation is composed of sums of kernels according to

$$S_Y(\underline{z}) = \sum_{\underline{y}\in Y} \mathcal{N}(\underline{z} - \underline{y}; \Gamma) \quad , \tag{5}$$

where Γ is a suitable kernel width and $\underline{z} \in \mathbb{R}^{d_y}$ is a free parameter. Intuitively, (5) interprets the measurements as a Gaussian mixture function, where the means coincide with the measurements. Also note that the order of the measurements in Y does not affect the result of $S_Y(\underline{z})$ in (5); it is a symmetric transformation, i.e., a set function.

The application of (5) to (1) yields to

$$\begin{aligned} S_{Y_k}(\underline{z}) &= S_{C_k}(\underline{z}) + S_{\{\bigcup_{1\leq l\leq N} \Theta_k^l\}}(\underline{z}) & (6)\\ &= S_{C_k}(\underline{z}) + \sum_{1\leq i\leq N} S_{\Theta_k^i}(\underline{z}) & (7)\\ &= S_{C_k}(\underline{z}) + \sum_{1\leq i\leq N} \boldsymbol{d}_k^i \cdot S_{\{\mathbf{H}_k^i \underline{\boldsymbol{x}}_k^i + \underline{\boldsymbol{v}}_k^i\}}(\underline{z}) \quad , & (8) \end{aligned}$$

where $d_k^i \in \{0,1\}$ indicates if the object is detected, i.e., $p(d_k^i = 1) = p_d$ and $p(d_k^i = 0) = 1 - p_d$.

The transformed measurement in (6) is scalar and still involves the free parameter $\underline{z}$. In order to get a sufficient number of measurement equations, we suggest to instantiate z with different test vectors $\underline{a}_k^1, \ldots, \underline{a}_k^{N_a}$, which yields

$$\underbrace{\begin{bmatrix} S_{Y_k}(\underline{a}_k^1) \\ \vdots \\ S_{Y_k}(\underline{a}_k^{N_a}) \end{bmatrix}}_{\underline{\boldsymbol{s}}_k} = \underbrace{\begin{bmatrix} S_{C_k}(\underline{a}_k^1) \\ \vdots \\ S_{C_k}(\underline{a}_k^{N_a}) \end{bmatrix}}_{\underline{\boldsymbol{c}}_k} + \begin{bmatrix} \sum_{1\leq i\leq N_a} \boldsymbol{d}_k^i S_{\{\mathbf{H}_k^i \underline{\boldsymbol{x}}_k^i + \underline{\boldsymbol{v}}_k^i\}}(\underline{a}_k^1) \\ \vdots \\ \sum_{1\leq i\leq N_a} \boldsymbol{d}_k^i S_{\{\mathbf{H}_k^i \underline{\boldsymbol{x}}_k^i + \underline{\boldsymbol{v}}_k^i\}}(\underline{a}_k^{N_a}) \end{bmatrix} . \tag{9}$$

Intuitively, the test vectors are supposed to lie close to the predicted target locations in order to capture the shape of the transformed measurements.

IV. Gaussian Estimator for the Kernel-SME

In the same manner as in [4], we derive a (nonlinear) Kalman filter for recursive estimation based on the measurement equation (9). For this purpose, we assume that the posterior probability density function of $\underline{\boldsymbol{x}}_k$ is Gaussian, i.e.,

$$p(\underline{x}_k \,|\, \mathbf{S}_k) = \mathcal{N}\left(\underline{x}_k; \underline{\mu}_k^x, \Sigma_k^x\right) \quad , \tag{10}$$

where $\underline{\mu}_k^x$ is the mean, Σ_k^x is the covariance matrix, and $\mathbf{S}_k = \{\underline{\boldsymbol{s}}_1, \ldots, \underline{\boldsymbol{s}}_k\}$ all pseudo-measurements up to time k.

1) Time Update: The time update step determines $p(\underline{x}_k \,|\, \mathbf{S}_{k-1}) = \mathcal{N}\left(\underline{x}_k; \underline{\mu}_{k|k-1}^x, \Sigma_{k|k-1}^x\right)$ based on the previous density $p(\underline{x}_{k-1} \,|\, \mathbf{S}_{k-1})$. Due to the linear system model, the prediction can be performed according the Kalman filter formulas

$$\begin{aligned} \underline{\mu}_{k|k-1}^x &= \mathbf{A}_k \cdot \underline{\mu}_{k-1}^x \quad \text{, and} & (11)\\ \Sigma_{k|k-1}^x &= \mathbf{A}_k \Sigma_{k-1}^x (\mathbf{A}_k)^T + \Sigma_k^w \quad , & (12) \end{aligned}$$

where Σ_k^w denotes the covariance matrix of the stacked system noise $\underline{\boldsymbol{w}}_k$.

2) Measurement Update: In the measurement update step, the prediction $\mathcal{N}\left(\underline{x}_k; \underline{\mu}_{k|k-1}^x, \Sigma_{k|k-1}^x\right)$ is updated with the pseudo-measurement $\underline{\boldsymbol{s}}_k$. For this purpose, we apply the Kalman filter formulas [2] to the measurement equation (9), i.e., the updated estimate becomes

$$\begin{aligned} \underline{\mu}_k^x &= \underline{\mu}_{k|k-1}^x + \Sigma_k^{xs} (\Sigma_k^{ss})^{-1} \left(\underline{\boldsymbol{s}}_k - \underline{\mu}_k^s\right) \quad \text{, and} & (13)\\ \Sigma_k^x &= \Sigma_{k|k-1}^x - \Sigma_k^{xs} (\Sigma_k^{ss})^{-1} \Sigma_k^{sx} , & (14) \end{aligned}$$

with

- predicted pseudo-measurement $\underline{\mu}_k^s$,
- cross-covariance Σ_k^{xs} between the state $\underline{\boldsymbol{x}}_k$ and pseudo-measurement $\underline{s}_k$, and
- covariance Σ_k^{ss} of the pseudo-measurement $\underline{s}_k$.

Closed-form expressions for the above moments can be derived similar to the case without missing and false measurements, see [4]. The detection indicator $\boldsymbol{d}_k$ can be incorporated in the moment calculations in a straightforward manner as it is mutually independent to all other involved random variables. Furthermore, the clutter measurements are independent of object-related measurements, i.e., $\underline{\boldsymbol{c}}_k$ is independent to the transformed object-related measurements. The following theorem derives the moments for the transformed clutter measurements.

Theorem 1. *Let* $C_k = \left\{ \underline{\boldsymbol{y}}_k^1, \ldots, \underline{\boldsymbol{y}}_k^{M_k^c} \right\}$ *in* (9) *be a set of* M_k^c *independent identically distributed random variables, where* M_k^c *is Poisson distributed with mean* $\lambda > 0$. *Then, the mean and covariance of the transformed clutter* $\underline{\boldsymbol{c}}_k$ *in* (9) *is given by*

$$
\begin{aligned}
\underline{\mu}_k^{c_i} &= \lambda \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(z) \Big\} \\
\Sigma_k^{c_i c_j} &= \lambda \, \mathrm{Cov}\{ S_{\underline{\boldsymbol{y}}_k}(\underline{a}_k^i), S_{\underline{\boldsymbol{y}}_k}(\underline{a}_k^j) \} + \\
&\qquad \lambda \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(\underline{a}_k^i) \Big\} \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(\underline{a}_k^j) \Big\}
\end{aligned}
$$

with $i, j \in \{1, \ldots, N_a\}$, *where* $\underline{\boldsymbol{y}}_k$ *has the same distribution as the random variables of* C_k.

Proof: Let $M_k^c \sim \mathrm{Bin}(n, p)$ be Binomial distributed with n trials of probability p. It is well known that the Binomial distribution converges to a Poisson distribution with $\lambda = np$ for $n \to \infty$ and $p \to 0$. Hence, we obtain for the mean

$$
\begin{aligned}
\mathrm{E}\{ S_{C_k}(z) \} &= \mathrm{E}\left\{ \sum_{i=1}^{n} \boldsymbol{d}_i S_{\underline{\boldsymbol{y}}_k^i}(z) \right\} \\
&= np \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(z) \Big\} \\
&\to \lambda \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(z) \Big\}
\end{aligned}
$$

and for the covariance

$$
\begin{aligned}
&\mathrm{Cov}\{ S_{C_k}(z_1), S_{C_k}(z_2) \} \\
&= \sum_{i=1}^{n} \mathrm{Cov}\{ \boldsymbol{d}_i S_{\underline{\boldsymbol{y}}_k^i}(z_1), \boldsymbol{d}_i S_{\underline{\boldsymbol{y}}_k^i}(z_2) \} \\
&= \sum_{i=1}^{n} \mathrm{E}\Big\{ \boldsymbol{d}_i^2 S_{\underline{\boldsymbol{y}}_k^i}(z_1) S_{\underline{\boldsymbol{y}}_k^i}(z_2) \Big\} - \mathrm{E}\Big\{ \boldsymbol{d}_i S_{\underline{\boldsymbol{y}}_k^i}(z_1) \Big\} \, \mathrm{E}\Big\{ \boldsymbol{d}_i S_{\underline{\boldsymbol{y}}_k^i}(z_2) \Big\} \\
&= \sum_{i=1}^{n} p_d \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k^i}(z_1) S_{\underline{\boldsymbol{y}}_k^i}(z_2) \Big\} - p_d^2 \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k^i}(z_1) \Big\} \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k^i}(z_2) \Big\} \\
&= np_d \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(z_1) S_{\underline{\boldsymbol{y}}_k}(z_2) \Big\} - np_d^2 \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(z_1) \Big\} \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(z_2) \Big\} \\
&\to \lambda \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(z_1) S_{\underline{\boldsymbol{y}}_k}(z_2) \Big\} \\
&= \lambda \, \mathrm{Cov}\{ S_{\underline{\boldsymbol{y}}_k}(z_1), S_{\underline{\boldsymbol{y}}_k}(z_2) \} + \lambda \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(z_1) \Big\} \, \mathrm{E}\Big\{ S_{\underline{\boldsymbol{y}}_k}(z_2) \Big\} \, .
\end{aligned}
$$

■

Fig. 1 depicts pseudo-code of the overall algorithm for the measurement update under the assumption that the object estimates are mutually uncorrelated, which is a standard assumptions in multi-object tracking, i.e., $\Sigma_k^{x_i x_j} = \mathbf{0}$ for all $i \neq j$. However, we note that this assumption is actually not necessary for the Kernel-SME filter; it just leads to simpler formulas, see also [4]. The overall run-time of the algorithm is $\mathcal{O}((N_k^a)^2 \cdot N + M_k)$, which is pretty efficient compared to other multi-object trackers. With further approximations, even more efficient algorithms could be achieved, e.g., assuming Σ_k^{ss} to be diagonal would lead to a linear-time algorithm in the number test-points and objects.

For each predicted object location, the pseudo-code in Fig. 1 selects two test vectors for each axis. In this manner, the number of test vectors does not depend on the number of measurements. More advanced methods for selecting test vectors might be reasonable. In general, an increasing number of test vectors leads to an increased performance of the Kernel-SME filter.

V. Evaluation

In order to assess the performance of the Kernel-SME filter, we consider a scenario with six two-dimensional target objects. The objects move according to the trajectory depicted in Fig. 2a. For the tracker, a nearly constant velocity model is employed, i.e., the state vector is

$$
\underline{\boldsymbol{x}}_k^l = \left[(\underline{\boldsymbol{p}}_k^l)^T, (\dot{\underline{\boldsymbol{p}}}_k^l)^T \right]^T \quad ,
$$

with $l \in \{1, \ldots, N\}$, where $\underline{\boldsymbol{p}}_k^l \in \mathbb{R}^2$ is the position and $\dot{\underline{\boldsymbol{p}}}_k^l \in \mathbb{R}^2$ the velocity. Furthermore, for the process model (3), we have

$$
\mathbf{A}_k^l = \begin{bmatrix} \mathbf{I}_2 & \mathbf{I}_2 \\ \mathbf{0}_2 & \mathbf{I}_2 \end{bmatrix}
$$

with two-dimensional identity matrix $\mathbf{I}_2$ and null matrix $\mathbf{0}_2$. The process noise covariance matrix is

$$
\Sigma_{k,l}^w = q_0 \begin{bmatrix} \frac{\mathbf{I}_2}{4} & \frac{\mathbf{I}_2}{2} \\ \frac{\mathbf{I}_2}{2} & \mathbf{I}_2 \end{bmatrix}
$$

with parameter $q_0 = 0.00004$.

At each time k, noisy position measurements are received, i.e., $\mathbf{H}_k^l = [\mathbf{I}_2, \mathbf{0}_2]$, with $\Sigma_{k,l}^v = 0.2\mathbf{I}_2$. The probability of detection is 0.95 and the number of false measurement is Poisson distributed with mean $\lambda = 7$, where the false measurements are uniformly distributed in the tracking area. Fig. 2b shows the overall measurements for an example run of 50 time steps. The Kernel-SME filter uses the kernel width $\Gamma = 1$.

The Kernel-SME filter falls into the category of computationally cheap and simple multi-object tracking algorithms. Hence, we compare the results with the Cheap JPDAF [6], which is a traditional fast approximation of the weights in the JPDAF filter. Both trackers are initialized close to the ground truth. For the comparison, we show the average OSPA error [16] for 50 Monte Carlo runs.

VI. Conclusions and Future Work

This paper is about the Kernel-SME filter [4] – a new method for multi-object tracking algorithm based on Kamen's Symmetric Measurement Equation (SME) idea. The Kernel-SME employs a fundamental new construction of SMEs that aims at solving the typical problems associated to the traditional SME approach. Specifically, in this paper we show how false and missing measurements can be incorporated in the Kernel-SME filter. This is a significant step forward as nearly all real-world applications come with false and missing measurements.

Input:

- Predicted states $\underline{\mu}_{k|k-1}^{x_1}, \ldots, \underline{\mu}_{k|k-1}^{x_N}$ of N objects with covariance matrices $\Sigma_{k|k-1}^{x_1}, \ldots, \Sigma_{k|k-1}^{x_N}$
- Measurements $\underline{\boldsymbol{y}}_k^1, \ldots, \underline{\boldsymbol{y}}_k^M$ (order of measurements is irrelevant)

Output:

- Updated state estimates $\underline{\mu}_k^{x_1}, \ldots, \underline{\mu}_k^{x_N}$ of N objects with covariance matrices $\Sigma_k^{x_1}, \ldots, \Sigma_k^{x_N}$

Algorithm:

1) Determine test vectors $\underline{a}_k^1, \ldots, \underline{a}_k^{N_a}$ with $N_a = 2 \cdot d_y \cdot N$ according to

$$\underline{a}_k^{l+i-1} := \mathbf{H}_k^l \underline{\mu}_{k|k-1}^{x_l} + \left(\sqrt{d_y \Gamma}\right)_i \quad \text{and} \quad \underline{a}_k^{l+2(i-1)} := \mathbf{H}_k^l \underline{\mu}_{k|k-1}^{x_l} - \left(\sqrt{d_y \Gamma}\right)_i$$

for $i = 1, \ldots, N$ and $l = 1, \ldots, d_y$, where $\left(\sqrt{d_y \Gamma}\right)_i$ denotes the i-th column of $\sqrt{d_y \Gamma}$.

2) Compute pseudo-measurement $\underline{\boldsymbol{s}}_k = \left[\underline{\boldsymbol{s}}_k^1, \ldots, \underline{\boldsymbol{s}}_k^{N_a}\right]^T$ with

$$\underline{\boldsymbol{s}}_k^i = \sum_{l=1}^{N} \mathcal{N}\left(\underline{a}_k^i; \underline{\boldsymbol{y}}_k^l, \Gamma\right) \text{ for } i = 1, \ldots, N_a$$

3) Determine $P_l^\Gamma(\underline{z}) := \mathcal{N}\left(\underline{z}; \mathbf{H}_k^l \underline{\mu}_l^x, \mathbf{H}_k^l \Sigma_{k|k-1}^{x_l} (\mathbf{H}_k^l)^T + \Sigma_k^v + \Gamma\right)$ for all $l = 1 \ldots N$

4) Determine moments of pseudo-measurements:

   a) Mean $\underline{\mu}_k^s = \left[\underline{\mu}_k^{s_1}, \ldots, \underline{\mu}_k^{s_{N_a}}\right]^T$ of predicted pseudo-measurement:

$$\underline{\mu}_{k,i}^s = p_d \cdot \sum_{l=1\ldots N} P_l^\Gamma(\underline{a}_k^i) + \underline{\mu}_k^{c_i} \text{ for } i = 1, \ldots, N_a$$

   b) Covariance $\Sigma_k^{ss} = (\Sigma_k^{s_i s_j})_{i,j=1,\ldots,N_a}$ of predicted pseudo-measurement:

$$\Sigma_k^{s_i s_j} = p_d \mathcal{N}\left(\underline{a}_k^i; \underline{a}_k^j, 2\Gamma\right) \cdot \sum_{l=1\ldots N} P_l^{0.5\Gamma}\left(\tfrac{1}{2}(\underline{a}_k^i + \underline{a}_k^j)\right) - p_d^2 P_l^\Gamma(\underline{a}_k^i) \cdot P_l^\Gamma(\underline{a}_k^j) + \Sigma_k^{c_i c_j}$$

5) Perform update for all objects $l = 1 \ldots N$:

   a) Cross-covariance $\Sigma_k^{x_l s} = \left[\Sigma_k^{x_l s_1}, \ldots, \Sigma_k^{x_l s_{N_a}}\right]$ between predicted pseudo-measurement:

$$\begin{aligned} \Sigma_k^{x_l s_i} &= -\underline{\mu}_k^{x_l} \cdot \underline{\mu}_k^{s_i} + p_d \cdot P_l^\Gamma(\underline{a}_k^i) \cdot (\underline{\mu}_{k|k-1}^{x_l} + \mathbf{K}_k^l(\underline{a}_k^i - \mathbf{H}_k^l \underline{\mu}_k^{x_l})) \text{ , where} \\ \mathbf{K}_k^l &= \Sigma_{k|k-1}^{x_l} \mathbf{H}_k^l \cdot \left(\mathbf{H}_k^l \Sigma_{k|k-1}^{x_l} (\mathbf{H}_k^l)^T + \Gamma + \Sigma_k^v\right)^{-1} \end{aligned}$$

   b) Kalman filter update

$$\begin{aligned} \underline{\mu}_k^{x_l} &= \underline{\mu}_{k|k-1}^{x_l} + \Sigma_k^{x_l s} (\Sigma_k^{ss})^{-1} \left(\underline{\boldsymbol{s}}_k - \underline{\mu}_k^s\right) \text{ , and} \\ \Sigma_k^{x_l} &= \Sigma_{k|k-1}^{x_l} - \Sigma_k^{x_l s} (\Sigma_k^{ss})^{-1} \Sigma_k^{s x_l} \ . \end{aligned}$$

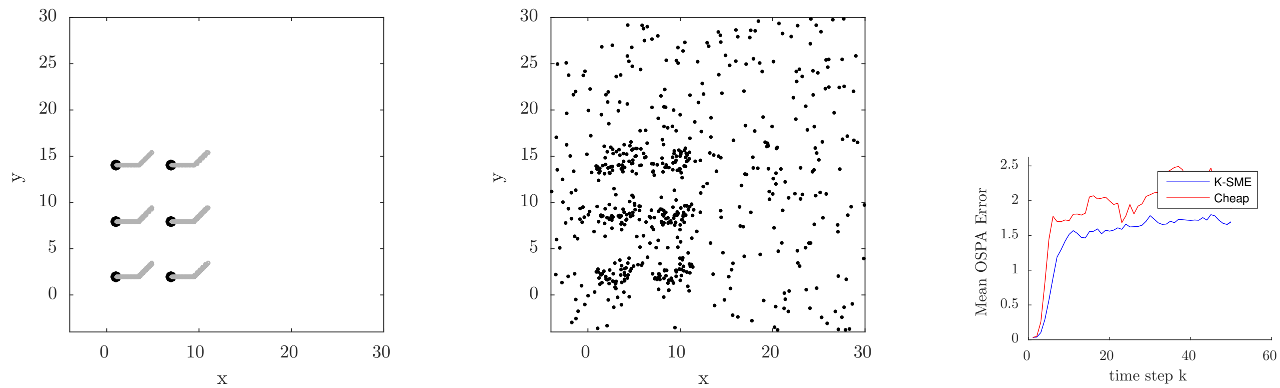
Fig. 1: Summary of the measurement update of the Kernel-SME filter.

The Kernel-SME filter is very efficient and involves rather simple formulas. Experiments show that the tracking quality can compete with traditional fast and simple multi-object tracking algorithms. Of course, the evaluation of multi-object tracking algorithms is in general quite complex and many different aspects have to be considered. Future evaluations and experiments will bring up the advantages and disadvantages of the Kernel-SME filter. Of course, due to its low computational demands, one cannot expect too much with regards to tracking quality. Obviously, the number and locations of the test vectors influences the quality of the Kernel-SME filter. Hence, future work shall analyze and discuss different methods for selecting test vectors. We also intend to extend the Kernel-SME filter to the case of an unknown number of objects and to the case of multiple measurements per object per time frame. Furthermore, Kernel-SME ideas have inspired new association-free multi-object tracking methods that are inherently association-free such as [7] that will be further investigated in the future.

REFERENCES

[1] Y. Bar-Shalom, F. Daum, and J. Huang, "The Probabilistic Data Association Filter," *IEEE Control Systems Magazine*, vol. 29, no. 6, pp. 82–100, December 2009.

[2] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: John Wiley & Sons, Inc., 2002.



(a) Initial object locations (big dots) and trajectory (small dots).

(b) Measurements from an example run.

(c) Mean OSPA error (for 50 runs).

Fig. 2: Evaluation: Kernel-SME vs. Cheap JPDAF.

[3] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, 2011.

[4] M. Baum and U. D. Hanebeck, “The Kernel-SME Filter for Multiple Target Tracking,” in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, Jul. 2013.

[5] M. Baum, B. Noack, and U. D. Hanebeck, “Kalman Filter-based SLAM with Unknown Data Association using Symmetric Measurement Equations,” in *Proceedings of the 2015 IEEE International Conference on Multisensor Fusion and Information Integration (MFI 2015)*, San Diego, California, USA, Sep. 2015.

[6] R. J. Fitzgerald, “Development of Practical PDA Logic for Multitarget Tracking by Microprocessor,” in *American Control Conference (ACC)*, 1986, pp. 889–898.

[7] U. D. Hanebeck and M. Baum, “Association-Free Direct Filtering of Multi-Target Random Finite Sets with Set Distance Measures,” in *Proceedings of the 18th International Conference on Information Fusion (Fusion 2015)*, Washington, USA, Jul. 2015.

[8] E. Kamen, “Multiple Target Tracking based on Symmetric Measurement Equations,” *IEEE Transactions on Automatic Control*, vol. 37, no. 3, pp. 371 –374, March 1992.

[9] E. Kamen and C. Sastry, “Multiple Target Tracking using Products of Position Measurements,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 2, pp. 476 –493, April 1993.

[10] G. Kurz, M. Baum, and U. D. Hanebeck, “Real-time Kernel-based Multiple Target Tracking for Robotic Beating Heart Surgery,” in *Proceedings of the Eighth IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM 2014)*, A Coruña, Spain, Jun. 2014.

[11] Y. J. Lee and E. W. Kamen, “SME Filter Approach to Multiple Target Tracking with False and Missing Measurements,” in *SPIE 1954, Signal and Data Processing of Small Targets*, 1993, pp. 574–586.

[12] W. Leven and A. Lanterman, “Multiple Target Tracking with Symmetric Measurement Equations using Unscented Kalman and Particle Filters,” in *Proceedings of the Thirty-Sixth Southeastern Symposium on System Theory*, 2004, pp. 195 – 199.

[13] ——, “Unscented Kalman Filters for Multiple Target Tracking With Symmetric Measurement Equations,” *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 370 –375, February 2009.

[14] R. P. S. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Boston, Mass.: Artech House, 2007.

[15] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House Publishers, 2004.

[16] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, “A Consistent Metric for Performance Evaluation of Multi-Object Filters,” *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447 –3457, Aug. 2008.

[17] Swati and S. Bhaumik, “Multiple Target Tracking based on Homogeneous Symmetric Transformation of Measurements,” *Aerospace Science and Technology*, 2012.

[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, Nov. 2006.

[19] F. Zhang, G. Hinz, D. Clarke, and A. Knoll, “Vehicle-infrastructure localization based on the SME filter,” in *IEEE 18th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2015, pp. 225–230.

[20] F. Zhang, G. Hinz, D. Gulati, D. Clarke, and A. Knoll, “Cooperative vehicle-infrastructure localization based on the symmetric measurement equation filter,” *GeoInformatica*, pp. 1–20, 2016.