

S²KF: The Smart Sampling Kalman Filter

Jannik Steinbring and Uwe D. Hanebeck

Intelligent Sensor-Actuator-Systems Laboratory (ISAS)

Institute for Anthropomatics

Karlsruhe Institute of Technology (KIT), Germany

jannik.steinbring@kit.edu, uwe.hanebeck@ieee.org

Abstract—An accurate Linear Regression Kalman Filter (LRKF) for nonlinear systems called Smart Sampling Kalman Filter (S²KF) is introduced. It is based on a new low-discrepancy Dirac Mixture approximation of Gaussian densities. The approximation comprises an arbitrary number of optimally and deterministically placed samples in the entire state space, so that the filter resolution can be adapted to either achieve high-quality results or meet computational constraints. For two samples per dimension, the S²KF comprises the UKF as a special case. With an increasing number of samples, the new filter quickly converges to the (typically infeasible) exact analytic LRKF. The S²KF can be seen as the ultimate generalization of all sample-based LRKFs such as the UKF, sigma-point filters, higher-order variants etc., as it homogeneously covers the state space with an arbitrary number of samples. It is evaluated by performing extended target tracking.

Index Terms—Nonlinear Kalman Filtering, LRKF, Dirac Mixtures, LCD, S²KF, Extended Object Tracking

I. INTRODUCTION

We consider estimating the hidden state of a discrete-time stochastic nonlinear dynamic system based on Bayesian inference. But, instead of maintaining the generally complex state densities, we focus on the assumption of a Gaussian distributed state vector in combination with Gaussian noise. However, even with these simplifications, closed-form solutions for the time, and especially for the measurement update, are rarely possible. For that reason, a further common step is to perform *statistical linearization* [1]. This makes it possible to perform backward inference without the need of an explicit likelihood function and use the well-known Kalman filter formulas instead. Filters based on statistical linearization are also referred to as *Linear Regression Kalman Filters* (LRKFs) [2], [1].

In case of linear systems corrupted by additive noise, the second Gaussian assumption, i.e., the statistical linearization, has no impact concerning estimation quality, and is reflected in the well-known Kalman filter equations [3]. But, in case of nonlinear systems, this assumption is often violated. The consequence is a diminished estimation performance compared to the more general Gaussian filters not making use of statistical linearization.

Basically, implementing an LRKF only amounts to calculating the first two moments of certain densities, depending on the given system and measurement equations. For some equations, these moments can be calculated analytically, including

polynomials, trigonometric functions, and their combinations [4]. Hence, this provides the LRKF with the best estimation quality. However, besides the problem of a restricted number of equations that allows analytic moment calculation itself, this approach requires an individual treatment of each occurring equation, which is time-consuming, error-prone, and prevents a generic filter, applicable to any system and measurement equation, regardless of its complexity.

A common solution are *sample-based* approaches, where the occurring state and noise densities are represented as a set of (non-)deterministically chosen samples. As a result, time and measurement update have to be adapted in order to handle these density representations. On the one hand, the samples have to be propagated individually through the given system and measurement equations. On the other hand, occurring analytic moment calculations have to be turned into their sample-based counterparts, i.e., sample mean and sample covariance. Of course, this introduces a further approximation step that may negatively affect the estimation performance. Nevertheless, employing a sample-based LRKF offers several advantages. First, due to the lack of an explicit use of a likelihood, the problem of sample degeneration is avoided¹, and, second, we obtain a generic filter that allows us to switch easily between different system and measurement equations without any additional effort. Moreover, this facilitates filter application engineering in the sense of rapid prototyping, as a newly designed system or measurement equation can be tried out immediately.

A. Contribution

In this paper, we introduce a new sample-based LRKF, which can be seen as the ultimate generalization of all sample-based LRKFs. For that purpose, we generate deterministic approximations of N -dimensional standard normal distributions comprising an arbitrary number of optimally placed samples in the entire state space. These sets of deterministically chosen samples serve as the fundamental basis for the new filter. In contrast to approaches using non-deterministic sampling, this lets the filter compute reproducible results and is more efficient, as a much smaller amount of samples has to be employed.

¹This is in contrast to filters explicitly using a likelihood, where backward inference implies a sample re-weighting that typically leads to a significantly reduced amount of samples contributing to the computation of the posterior moments, and consequently, to inaccurate results.

By simply increasing the number of employed samples, the new filter converges to the analytic LRKF, as the resulting approximation of the standard normal distribution represents higher order moments more accurately. Moreover, this approach requires only a single, especially simple to understand, optimization parameter. This makes filter fine-tuning simple, even for people not very familiar with (sample-based) Kalman filtering.

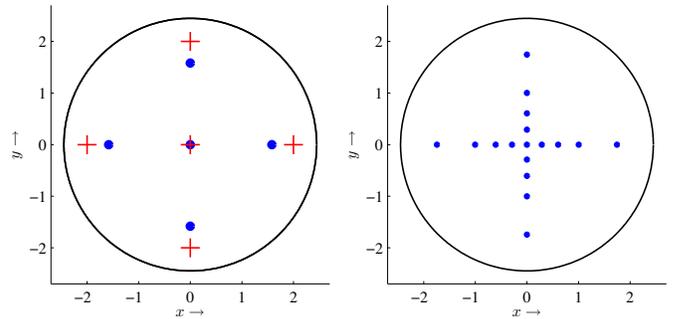
B. Related Work

One of the most popular sample-based LRKFs is the *Unscented Kalman Filter* (UKF) [5], [6]. It employs $2N + 1$ systematically chosen, axis-aligned samples for the time and measurement update, where N denotes the sum of state and noise dimensions (see Figure 1a). One of its greatest advantages is the ease with which the sample set can be created as well as the low computational effort due to the small amount of used samples. However, this last property introduces its main drawbacks. First, the UKF is hardly capable of approximating moments of a Gaussian distribution higher than second order. Second, the state space coverage suffers from the fact that the samples are placed solely on the principal axes. Both of these factors have a negative impact on the estimation quality. And third, the small amount of employed samples makes it rather likely to compute non-positive definite, i.e., invalid, covariance matrices, and thus, makes it hard for filtering applications to work reliably. For example, if all samples fall onto the roots of a sinus-shaped function [7].

Another drawback of the UKF are its rather unintuitive parameters that control the sample spread, i.e., the scaling, and their weighting (see Figure 1a). Besides the use of heuristics for determining these parameters, maximum likelihood estimators can be employed for this. In [8], the authors select a limited set of possible values for the scaling parameter. During a filter step they perform a measurement update for all selected scaling values individually, and choose the update which fits best with the given measurement as the actual filter step. Instead of simply trying various parameters during a filter step, in [7] the authors propose a parameter determination based on a Gaussian process optimization. Both approaches can improve the estimation quality, but also introduce new parameters (the possible scaling values and parameters controlling the optimization) that have to be determined in some way. Moreover, despite the additional computational effort, due to the several computed updates for one filter step, the number of samples remains the same, and, hence, the problems of state space coverage and covariance positive-definiteness are left unchanged.

The *Gaussian Filter* (GF) [9] improves the sampling by deterministically placing an arbitrary number of samples on each principal axis (see Figure 1b). Although the number of samples can easily be adjusted, and the covariance computation is more reliable, the state space coverage still remains sparse.

A non-deterministic LRKF sampling approach is given in [10], where the moments for a filter step are calculated with the aid of an iterative stochastic integration rule, where each iteration uses an additional UKF sample set with random



(a) UKF using two scalings.

(b) GF using 17 samples.

Figure 1: Sampling methods of a two-dimensional standard normal distribution. Covariance matrices with confidence interval of 95% (black lines). UKF with scaling 0.5 (blue points) and scaling 1.5 (red crosses).

scaling and rotation. Even though no complex parameters are required and the state space coverage is improved, the approach relies on the law of large numbers, and, hence, requires, even in larger state spaces, a large amount of samples to produce satisfying estimation results. In addition, estimation results are not reproducible due to its non-deterministic manner.

In order to improve overall LRKF estimation quality, a mixture of analytic and sample-based moment calculation (semi-analytic approach) should also be used, if possible [4].

In contrast to statistical linearization, an explicit linearization based on Taylor series approximation is also possible, such as done by the *Extended Kalman Filter* (EKF) [11] or its second-order variants [12]. The *Divided Difference Filter* (DDF) [13] based on polynomial and derivative-free approximations by employing a multivariable extension of Stirling's interpolation formula.

C. Overview

The remainder of this paper is structured as follows. In Sec. II, we give a detailed formulation of the filtering problem using statistical linearization. Sec. III describes the general sample-based LRKF approach. After that, in Sec. IV, we introduce our new *Smart Sampling Kalman Filter* (S^2KF). Extensive evaluation of the new filter is performed in Sec. V. The conclusion and potential future work concerning the new filter is presented in Sec. VI. Finally, in Sec. VII, links to online resources concerning the new filter and this paper are provided.

II. PROBLEM FORMULATION

We consider estimating the hidden state \mathbf{x}_k of a discrete-time stochastic nonlinear dynamic system with system equation

$$\mathbf{x}_k = \mathbf{a}_k(\mathbf{x}_{k-1}, \hat{\mathbf{u}}_k, \mathbf{w}_k), \quad (1)$$

and measurement equation

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k, \hat{\mathbf{r}}_k, \mathbf{v}_k), \quad (2)$$

where the subscript k denotes the discrete time step, $\hat{\underline{u}}_k$ and $\hat{\underline{r}}_k$ known inputs, and \underline{w}_k as well as \underline{v}_k known Gaussian noise processes according to

$$\underline{w}_k \sim \mathcal{N}(\underline{w}_k, \hat{\underline{u}}_k, \mathbf{C}_k^w)$$

and

$$\underline{v}_k \sim \mathcal{N}(\underline{v}_k, \hat{\underline{r}}_k, \mathbf{C}_k^v), \quad (3)$$

with means $\hat{\underline{u}}_k$ and $\hat{\underline{r}}_k$, and covariance matrices \mathbf{C}_k^w and \mathbf{C}_k^v , respectively. It is assumed that both noise processes are mutually independent and also independent of the state. For brevity, the inputs $\hat{\underline{u}}_k$ and $\hat{\underline{r}}_k$ are omitted for the remainder of this paper.

Furthermore, we restrict ourselves to a Gaussian distributed state \underline{x}_k for all time steps. We denote the estimated state distribution at time step k after incorporating k given measurements $\tilde{\underline{y}}_1, \dots, \tilde{\underline{y}}_{k-1}, \tilde{\underline{y}}_k$ as

$$\begin{aligned} f_k^e(\underline{x}_k) &= f(\underline{x}_k | \tilde{\underline{y}}_k, \tilde{\underline{y}}_{k-1}, \dots, \tilde{\underline{y}}_1) = f(\underline{x}_k | \tilde{\underline{y}}_{k:1}) \\ &= \mathcal{N}(\underline{x}_k, \hat{\underline{x}}_k, \mathbf{C}_k^e). \end{aligned}$$

Our goal is to perform a recursive state estimation based on Bayesian inference. A Bayesian estimator consists of two alternating steps, the time update, and the measurement update.

A. Time Update

The objective of the time update, also called prediction step, is to propagate the last known state estimation $f_{k-1}^e(\underline{x}_{k-1})$ (from the past) to the present by exploiting the given system model (1) in the form of its transition density $f_k^a(\underline{x}_k | \underline{x}_{k-1})$. This yields the *predicted* state estimation $f_k^p(\underline{x}_k)$ according to the CHAPMAN-KOLOMOGOROV equation

$$\begin{aligned} f_k^p(\underline{x}_k) &= f(\underline{x}_k | \tilde{\underline{y}}_{k-1}, \tilde{\underline{y}}_{k-2}, \dots, \tilde{\underline{y}}_1) = f(\underline{x}_k | \tilde{\underline{y}}_{k-1:1}) \\ &= \int f_k^a(\underline{x}_k | \underline{x}_{k-1}) \cdot f_{k-1}^e(\underline{x}_{k-1}) d\underline{x}_{k-1} \\ &= \iint \delta(\underline{x}_k - \underline{a}_k(\underline{x}_{k-1}, \underline{w}_k)) \cdot \\ &\quad f_{k-1}^e(\underline{x}_{k-1}) \cdot f_k^w(\underline{w}_k) d\underline{x}_{k-1} d\underline{w}_k, \end{aligned} \quad (4)$$

where $\delta(\cdot)$ denotes the Dirac delta function [11]. However, even though the prior state density $f_{k-1}^e(\underline{x}_{k-1})$ is Gaussian, this holds, in general, not for the predicted state density $f_k^p(\underline{x}_k)$. Therefore, we have to perform a subsequent moment matching in order to fulfill our Gaussian state assumption by computing mean

$$\hat{\underline{x}}_k^p = \int \underline{x}_k \cdot f_k^p(\underline{x}_k) d\underline{x}_k \quad (5)$$

and covariance

$$\mathbf{C}_k^p = \int (\underline{x}_k - \hat{\underline{x}}_k^p) \cdot (\underline{x}_k - \hat{\underline{x}}_k^p)^T \cdot f_k^p(\underline{x}_k) d\underline{x}_k \quad (6)$$

of $f_k^p(\underline{x}_k)$, and finally approximating the predicted state density according to

$$f_k^p(\underline{x}_k) \approx \mathcal{N}(\underline{x}_k, \hat{\underline{x}}_k^p, \mathbf{C}_k^p).$$

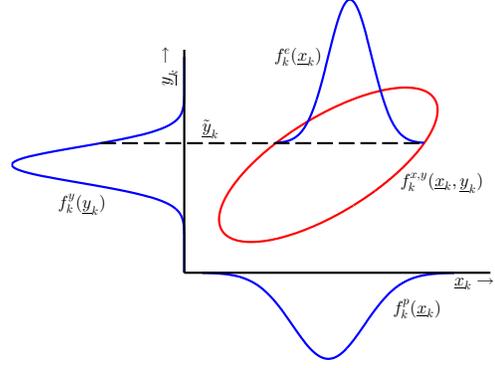


Figure 2: Statistical linearization in the case of an one-dimensional state \underline{x}_k and measurement \underline{y}_k . The given measurement $\tilde{\underline{y}}_k$ determines the position of where to slice the joint density (dashed line) in order to get the posterior state estimation $f_k^e(\underline{x}_k)$. It should be noted that the mean of posterior state density is different from that of the prior due to the existing correlation between state and measurement as well as the difference between the most expected measurement $\hat{\underline{y}}_k$ and the given measurement $\tilde{\underline{y}}_k$.

B. Measurement Update

The measurement update or filter step incorporates a given measurement $\tilde{\underline{y}}_k$ at time step k into the predicted state estimation $f_k^p(\underline{x}_k)$ in order to correct it.

By turning the measurement model (2) into its corresponding likelihood $f_k^h(\underline{y}_k = \tilde{\underline{y}}_k | \underline{x}_k)$ by assuming that the current measurement \underline{y}_k is conditionally independent of the prior measurements given the current state estimation, the corrected state estimation can be obtained according to

$$\begin{aligned} f_k^e(\underline{x}_k) &= f(\underline{x}_k | \tilde{\underline{y}}_{k:1}) \\ &= \frac{f_k^h(\underline{y}_k = \tilde{\underline{y}}_k | \underline{x}_k) \cdot f_k^p(\underline{x}_k)}{f(\underline{y}_k = \tilde{\underline{y}}_k | \tilde{\underline{y}}_{k-1:1})} \\ &= \frac{f_k^{x,y}(\underline{x}_k, \underline{y}_k = \tilde{\underline{y}}_k | \tilde{\underline{y}}_{k-1:1})}{f(\underline{y}_k = \tilde{\underline{y}}_k | \tilde{\underline{y}}_{k-1:1})}, \end{aligned} \quad (7)$$

where $f(\underline{y}_k = \tilde{\underline{y}}_k | \tilde{\underline{y}}_{k-1:1})$ is a normalization constant. As we force $f_k^e(\underline{x}_k)$ to be Gaussian, its computation can be simplified by making the additional, rough assumption in (7) of a Gaussian distributed joint density $f_k^{x,y}(\cdot, \cdot)$ of prior state \underline{x}_k and measurement \underline{y}_k according to

$$f_k^{x,y}(\cdot, \cdot) \approx \mathcal{N}\left(\begin{bmatrix} \underline{x}_k \\ \underline{y}_k \end{bmatrix}, \begin{bmatrix} \hat{\underline{x}}_k^p \\ \hat{\underline{y}}_k \end{bmatrix}, \begin{bmatrix} \mathbf{C}_k^p & \mathbf{C}_k^{x,y} \\ (\mathbf{C}_k^{x,y})^T & \mathbf{C}_k^y \end{bmatrix}\right),$$

where $\hat{\underline{y}}_k$ and \mathbf{C}_k^y denote the measurement mean and covariance, and $\mathbf{C}_k^{x,y}$ the cross-covariance matrix of state and measurement [11], [1]. This assumption of a second Gaussian density, related to the already assumed Gaussian distributed state, is called statistical linearization [1], as the relation between state and measurement, i.e., the measurement model (2), gets linearized by incorporating the entire statistical information of the prior state estimation $f_k^p(\underline{x}_k)$.

This simplification allows us to compute the corrected state mean and covariance analytically by using the Kalman filter formulas [11]

$$\hat{\underline{x}}_k^e = \hat{\underline{x}}_k^p + \mathbf{C}_k^{x,y} \cdot (\mathbf{C}_k^y)^{-1} \cdot (\tilde{\underline{y}}_k - \hat{\underline{y}}_k) , \quad (8)$$

and

$$\mathbf{C}_k^e = \mathbf{C}_k^p - \mathbf{C}_k^{x,y} \cdot (\mathbf{C}_k^y)^{-1} \cdot (\mathbf{C}_k^{x,y})^T , \quad (9)$$

and set the corrected state estimation to

$$f_k^e(\underline{x}_k) : \approx \mathcal{N}(\underline{x}_k, \hat{\underline{x}}_k^e, \mathbf{C}_k^e)$$

afterwards. Based on the given measurement model (2), measurement noise (3), and predicted state estimation $f_k^p(\underline{x}_k)$, we can compute the required measurement mean

$$\begin{aligned} \hat{\underline{y}}_k &= \int \underline{y}_k \cdot f_k^y(\underline{y}_k) \, d\underline{y}_k \\ &= \iint \underline{h}_k(\underline{x}_k, \underline{v}_k) \cdot f_k^p(\underline{x}_k) \cdot f_k^v(\underline{v}_k) \, d\underline{x}_k \, d\underline{v}_k , \end{aligned} \quad (10)$$

measurement covariance

$$\begin{aligned} \mathbf{C}_k^y &= \int (\underline{y}_k - \hat{\underline{y}}_k) \cdot (\underline{y}_k - \hat{\underline{y}}_k)^T \cdot f_k^y(\underline{y}_k) \, d\underline{y}_k \\ &= \iint \underline{h}_k(\underline{x}_k, \underline{v}_k) \cdot \underline{h}_k(\underline{x}_k, \underline{v}_k)^T \cdot \\ &\quad f_k^p(\underline{x}_k) \cdot f_k^v(\underline{v}_k) \, d\underline{x}_k \, d\underline{v}_k - \hat{\underline{y}}_k \cdot \hat{\underline{y}}_k^T , \end{aligned} \quad (11)$$

and state measurement cross-covariance

$$\begin{aligned} \mathbf{C}_k^{x,y} &= \iint (\underline{x}_k - \hat{\underline{x}}_k^p) \cdot (\underline{y}_k - \hat{\underline{y}}_k)^T \cdot \\ &\quad f_k^{x,y}(\underline{x}_k, \underline{y}_k) \, d\underline{x}_k \, d\underline{y}_k \\ &= \iint \underline{x}_k \cdot \underline{h}_k(\underline{x}_k, \underline{v}_k)^T \cdot \\ &\quad f_k^p(\underline{x}_k) \cdot f_k^v(\underline{v}_k) \, d\underline{x}_k \, d\underline{v}_k - \hat{\underline{x}}_k^p \cdot \hat{\underline{y}}_k^T . \end{aligned} \quad (12)$$

The resulting measurement distribution

$$f_k^y(\underline{y}_k) = \mathcal{N}(\underline{y}_k, \hat{\underline{y}}_k, \mathbf{C}_k^y)$$

can be seen as which measurements the predicted state $f_k^p(\underline{x}_k)$ most likely expects at time step k .

A measurement update using statistical linearization can be illustrated as a given measurement $\tilde{\underline{y}}_k$ slices the joint probability density $f_k^{x,y}(\cdot, \cdot)$ at a certain position in order to obtain the posterior Gaussian state $f_k^e(\underline{x}_k)$ (see Figure 2). It is important to note that this measurement update is much different from performing the unmodified measurement update using the likelihood $f_k^h(\underline{y}_k = \tilde{\underline{y}}_k | \underline{x}_k)$ and reduce $f_k^e(\underline{x}_k)$ to a Gaussian afterwards.

C. Recursive State Estimation

The alternating use of the introduced time and measurement update, together with a given initial state estimation $f_0^e(\underline{x}_0)$, forms the desired recursive estimation process called Linear Regression Kalman Filter. Algorithm 1 summarizes the general LRKF procedure.

Algorithm 1 Linear Regression Kalman Filter

- 1: Set $f_0^e(\underline{x}_0) = \mathcal{N}(\underline{x}_0, \hat{\underline{x}}_0, \mathbf{C}_0)$
 - 2: **for** $k = 1, 2, \dots$ **do**
 - Time Update:**
 - 3: Compute predicted state moments $\hat{\underline{x}}_k^p$ and \mathbf{C}_k^p according to (5) and (6)
 - 4: Set $f_k^p(\underline{x}_k) = \mathcal{N}(\underline{x}_k, \hat{\underline{x}}_k^p, \mathbf{C}_k^p)$
 - 5: **if** measurement $\tilde{\underline{y}}_k$ is available **then**
 - Measurement Update:**
 - 6: Compute measurement moments $\hat{\underline{y}}_k$, \mathbf{C}_k^y , and $\mathbf{C}_k^{x,y}$ according to (10), (11), and (12)
 - 7: Compute posterior state moments $\hat{\underline{x}}_k^e$ and \mathbf{C}_k^e according to (8) and (9)
 - 8: Set $f_k^e(\underline{x}_k) = \mathcal{N}(\underline{x}_k, \hat{\underline{x}}_k^e, \mathbf{C}_k^e)$
 - 9: **else**
 - 10: Set $f_k^e(\underline{x}_k) = f_k^p(\underline{x}_k)$
 - 11: **end if**
 - 12: **end for**
-

III. SAMPLE-BASED LRKF

Calculating the required moments for the time and the measurement update analytically provides the LRKF with the best estimation quality. But, in case of non-existent closed-form solutions or complicated system and measurement equations, an approximate moment calculation has to be performed.

One way to achieve this is to replace the occurring state and noise densities with proper *Dirac Mixture* densities, that is, sample-based representations. As only a limited number of samples can be used, this replacement always entails a density approximation, and, hence, leads to an, in general, inferior LRKF compared to the analytic solutions.

A Dirac Mixture approximation of an arbitrary density function $f(\underline{s})$, encompassing L samples, is defined as [14], [4]

$$f(\underline{s}) \approx \sum_{i=1}^L w_i \cdot \delta(\underline{s} - \underline{s}_i) ,$$

with samples \underline{s}_i and non-negative scalar sample weights w_i , which hold

$$\sum_{i=1}^L w_i = 1 .$$

The samples \underline{s}_i can be chosen in a random or deterministic fashion. Moreover, a combination of both techniques is also possible.

A. Sample-Based Time Update

Our goal is compute the required moments (5) and (6) for the LRKF time update based on Dirac Mixtures. Therefore, we have to replace the density product $f_{k-1}^e(\underline{x}_{k-1}) \cdot f_k^w(\underline{w}_k)$ with an appropriate Dirac Mixture. Of course, each density could be approximated separately and the product of the resulting Dirac Mixtures built afterwards.

However, this would cause two drawbacks. First, the result of this density product would be the Cartesian product of the employed state and noise Dirac Mixtures, i.e., a Dirac Mixture with $L \cdot M$ samples, where L and M denote the respective number of samples of the state and noise Dirac Mixtures. Such an approach would not scale efficiently with an increasing number of employed samples. Second, state \underline{x}_{k-1} and noise \underline{w}_k are independent of each other and their respective densities, $f_{k-1}^e(\underline{x}_{k-1})$ and $f_k^w(\underline{w}_k)$, are Gaussian. Thus, the product is equivalent to their, also Gaussian, joint density $f_k^{x,w}(\underline{x}_{k-1}, \underline{w}_k)$ with a zero cross-covariance matrix $\mathbf{C}_k^{x,w}$. As a consequence, in case of separate approximations, the resulting joint density with its grid-like placed samples would not be optimal for approximating the ellipsoid Gaussian distribution $f_k^{x,w}(\underline{x}_{k-1}, \underline{w}_k)$.

Hence, we can do better by approximating the joint density directly according to

$$\begin{aligned} f_k^{x,w}(\underline{x}_{k-1}, \underline{w}_k) &= f_{k-1}^e(\underline{x}_{k-1}) \cdot f_k^w(\underline{w}_k) \\ &= \mathcal{N}\left(\begin{bmatrix} \underline{x}_{k-1} \\ \underline{w}_k \end{bmatrix}, \begin{bmatrix} \hat{\underline{x}}_{k-1} \\ \hat{\underline{w}}_k \end{bmatrix}, \begin{bmatrix} \mathbf{C}_{k-1}^e & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_k^w \end{bmatrix}\right) \\ &\approx \sum_{i=1}^{L_k^p} w_{k,i}^p \cdot \delta\left(\begin{bmatrix} \underline{x}_{k-1} \\ \underline{w}_k \end{bmatrix} - \begin{bmatrix} \underline{x}_{k-1,i} \\ \underline{w}_{k,i} \end{bmatrix}\right). \end{aligned} \quad (13)$$

Plugging this into (4) and exploiting the Dirac sifting property, yields to the predicted Dirac Mixture state density

$$\begin{aligned} f_k^p(\underline{x}_k) &\approx \sum_{i=1}^{L_k^p} w_{k,i}^p \cdot \delta(\underline{x}_k - \underline{a}_k(\underline{x}_{k-1,i}, \underline{w}_{k,i})) \\ &= \sum_{i=1}^{L_k^p} w_{k,i}^p \cdot \delta(\underline{x}_k - \underline{a}_{k,i}), \end{aligned} \quad (14)$$

with new samples $\underline{a}_{k,i}$ and unchanged sample weights $w_{k,i}^p$. The desired moments can now be obtained by computing sample mean

$$\hat{\underline{x}}_k^p \approx \sum_{i=1}^{L_k^p} w_{k,i}^p \cdot \underline{a}_{k,i}, \quad (15)$$

and sample covariance

$$\mathbf{C}_k^p \approx \sum_{i=1}^{L_k^p} w_{k,i}^p \cdot (\underline{a}_{k,i} - \hat{\underline{x}}_k^p) \cdot (\underline{a}_{k,i} - \hat{\underline{x}}_k^p)^T \quad (16)$$

of (14).

B. Sample-Based Measurement Update

The sample-based measurement update can be computed in a similar manner. We approximate the joint density $f_k^{x,v}(\underline{x}_k, \underline{v}_k)$ of state and measurement noise according to

$$\begin{aligned} f_k^{x,v}(\underline{x}_k, \underline{v}_k) &= f_k^p(\underline{x}_k) \cdot f_k^v(\underline{v}_k) \\ &= \mathcal{N}\left(\begin{bmatrix} \underline{x}_k \\ \underline{v}_k \end{bmatrix}, \begin{bmatrix} \hat{\underline{x}}_k^p \\ \hat{\underline{v}}_k \end{bmatrix}, \begin{bmatrix} \mathbf{C}_k^p & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_k^v \end{bmatrix}\right) \\ &\approx \sum_{i=1}^{L_k^e} w_{k,i}^e \cdot \delta\left(\begin{bmatrix} \underline{x}_k \\ \underline{v}_k \end{bmatrix} - \begin{bmatrix} \underline{x}_{k,i} \\ \underline{v}_{k,i} \end{bmatrix}\right), \end{aligned} \quad (17)$$

and plug this into (10), (11), and (12). This yields the desired measurement mean

$$\hat{\underline{y}}_k = \sum_{i=1}^{L_k^e} w_{k,i}^e \cdot \underline{h}_k(\underline{x}_{k,i}, \underline{v}_{k,i}), \quad (18)$$

measurement covariance

$$\begin{aligned} \mathbf{C}_k^y &= \sum_{i=1}^{L_k^e} w_{k,i}^e \cdot \underline{h}_k(\underline{x}_{k,i}, \underline{v}_{k,i}) \cdot \\ &\quad \underline{h}_k(\underline{x}_{k,i}, \underline{v}_{k,i})^T - \hat{\underline{y}}_k \cdot \hat{\underline{y}}_k^T, \end{aligned} \quad (19)$$

and state measurement cross-covariance

$$\mathbf{C}_k^{x,y} = \sum_{i=1}^{L_k^e} w_{k,i}^e \cdot \underline{x}_{k,i} \cdot \underline{h}_k(\underline{x}_{k,i}, \underline{v}_{k,i})^T - \hat{\underline{x}}_k^p \cdot \hat{\underline{y}}_k^T. \quad (20)$$

In summary, performing the sample-based time and measurement update is equivalent to propagating the samples from the joint densities (13) and (17) individually through the system and measurement equation (1) and (2), and subsequently computing the sample means (15) and (18), and the sample covariances (16), (19), and (20).

IV. THE SMART SAMPLING KALMAN FILTER

Sec. III dealt with the general sample-based LRKF. In order to use it, appropriate Dirac Mixture approximations of Gaussian densities to have be determined, i.e., sets of samples and their respective sample weights.

We focus on deterministic Dirac Mixtures and utilize a sample generator based on the *Localized Cumulative Distribution* (LCD) described in [15], [16]. The LCD approach turns the density approximation problem into an optimization problem. More precisely, for a given number of samples, it determines their optimal positions *in the entire state space* by systematically minimizing a modified Cramér-von Mises distance between the Dirac Mixture approximation and an arbitrary Gaussian distribution.

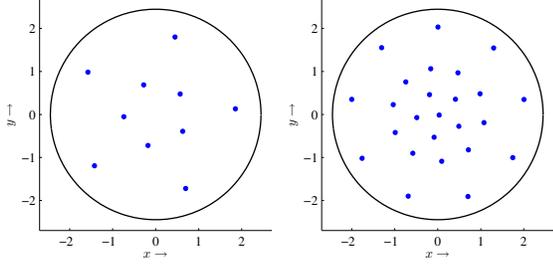
Even though the LCD approach can approximate an arbitrary Gaussian, it is computationally expensive, and thus, not well suited for online usage. Hence, we have to be content with an offline generated Dirac Mixture approximation

$$f_{\text{LCD}}(\underline{s}) = \frac{1}{L} \sum_{i=1}^L \delta(\underline{s} - \underline{s}_i) \approx \mathcal{N}(\underline{s}, \mathbf{I}), \quad (21)$$

of a standard normal distribution with L equally weighted, that is, $w_i = \frac{1}{L}$, and optimally placed samples \underline{s}_i , where L can be chosen arbitrarily.

As the LRKF explicitly relies on the first two moments of the employed densities, it is indispensable that the LCD-generated Dirac Mixture approximation of a standard normal distribution (21) has a zero sample mean, that is

$$\hat{\underline{s}} = \frac{1}{L} \sum_{i=1}^L \underline{s}_i = \underline{0},$$



(a) S²KF using 10 samples. (b) S²KF using 25 samples.

Figure 3: LCD sampling of a two-dimensional standard normal distribution. Covariance matrices with confidence interval of 95% (black lines). The excellent state space coverage can be clearly seen.

and a unit sample covariance

$$\mathbf{C}^s = \frac{1}{L} \sum_{i=1}^L (\mathbf{s}_i - \hat{\mathbf{s}}) \cdot (\mathbf{s}_i - \hat{\mathbf{s}})^T = \mathbf{I}_L ,$$

where \mathbf{I}_L denotes the L -dimensional identity matrix. Figures 3a and 3b depict such generated Dirac Mixture approximations for two different numbers of samples in case of a two-dimensional standard normal distribution.

Given an arbitrary Gaussian distribution

$$g(\mathbf{z}) = \mathcal{N}(\mathbf{z}, \hat{\mathbf{z}}, \mathbf{C}^z)$$

during filter usage, i.e., the densities $f_k^{x,w}(\mathbf{x}_{k-1}, \mathbf{w}_k)$ and $f_k^{x,v}(\mathbf{x}_k, \mathbf{v}_k)$, we compute the matrix square root $\sqrt{\mathbf{C}^z}$ of \mathbf{C}^z using the Cholesky decomposition², and individually rotate and scale each sample of (21) according to

$$\mathbf{z}_i := \sqrt{\mathbf{C}^z} \cdot \mathbf{s}_i + \hat{\mathbf{z}} \quad \forall i \in \{1, \dots, L\} , \quad (22)$$

where the samples \mathbf{z}_i represent the transformed Dirac Mixture approximation.

By employing the LCD-generated Dirac Mixture approximation (21) in combination with the transformation (22) during the sample-based LRKF time and measurement update, we introduce the new *Smart Sampling Kalman Filter* (S²KF), with its powerful feature of *using an arbitrary number of optimally placed samples*. There exist no sampling constraints such as axis aligned samples or that the number of samples must be a multiple of the state dimension as with the UKF or GF.

With an increasing number of used samples in (21), the S²KF quickly converges to the analytic LRKF, as the resulting Dirac Mixture approximation of the standard normal distribution quickly becomes more accurate, that is, captures more higher order moments. In contrast to the UKF with its fixed-size sample set, this allows an extensive evaluation of the given system and measurement models, as more and more samples become available in the important regions of the state space. Moreover, this makes a non-positive definite state covariance

²Other matrix square root operations, such as the eigendecomposition, are also possible.

matrix more unlikely and the filter more reliably. As a consequence, the estimation quality can be easily improved by simply increasing the number of employed samples, which offers an *intuitive and easy-to-use optimization parameter*. Of course, this effect is also true for filters relying on random sampling. But, due to the optimal sample placement, the S²KF converges much faster, so that already a small number of samples provides an excellent estimation quality.

As the standard normal distribution is rotation-invariant, the LCD generator cannot create unique Dirac Mixture approximations (21) for a given dimension of \underline{s} and number of samples L . Furthermore, for each combination of dimension of \underline{s} and number of samples L , an own Dirac Mixture approximation is required, as the LCD-generator performs a global optimization that takes all sample positions into account. Hence, a once created Dirac Mixture cannot be extended by simply adding additional samples and leaving the existing samples unchanged.

But, if we reuse a once generated LCD approximation, the filter results become reproducible. For that reason, we store each generated LCD approximation persistent in the file system for later reuse. We call this storage the *Sample Cache*. Additionally, if a required sample set for a given combination of dimension and number of samples is not yet available in the Sample Cache during filter usage, it is generated on demand, that is, *transparent for the user*, and subsequently stored in Sample Cache. A positive side-effect is that this avoids a complete regeneration of all required sample sets on each program start, and over time, the Sample Cache grows and the necessity for time-consuming sample generation becomes more unlikely during filter usage.

V. EVALUATION

In this section, we evaluate the S²KF by means of tracking an extended target modeled as multiplicative noise.

Our goal is to estimate the position $\mathbf{c}_k = [\mathbf{c}_k^x, \mathbf{c}_k^y]^T$ and extent $\mathbf{l}_k = [\mathbf{l}_k^x, \mathbf{l}_k^y]^T$ of a two-dimensional axis-aligned rectangular target (see Figure 4a). The hidden system state is given by $\mathbf{x}_k = [\mathbf{l}_k^T, \mathbf{c}_k^T]^T$.

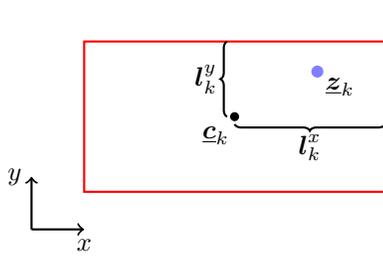
In order to incorporate target information into our state estimation, we assume uniformly distributed, noisy measurements stemming from the surface of the target. For this purpose, we extend the approach proposed in [17]. The basic idea is that each point of the target surface can be reached by scaling the axis lengths \mathbf{l}_k^x and \mathbf{l}_k^y individually and adding the center \mathbf{c}_k , i.e.,

$$\mathbf{z}_k = \mathbf{H} \cdot \mathbf{l}_k + \mathbf{c}_k ,$$

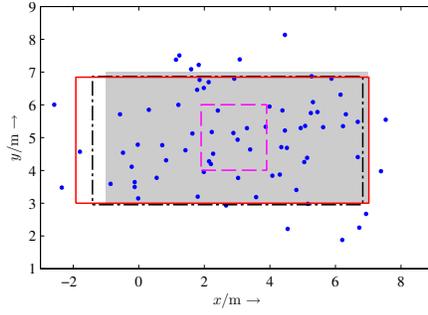
with uncorrelated state independent multiplicative noise $\mathbf{H} = \text{diag}(\mathbf{h}^x, \mathbf{h}^y)$. As the measurements are uniformly distributed, \mathbf{h}^x and \mathbf{h}^y also have to be uniformly distributed in the interval $[-1, 1]$ (see Figure 4a). Taking the measurement noise into account yields the preliminary nonlinear measurement equation

$$\mathbf{m}_k = \mathbf{z}_k + \mathbf{w}_k = \mathbf{H} \cdot \mathbf{l}_k + \mathbf{c}_k + \mathbf{w}_k , \quad (23)$$

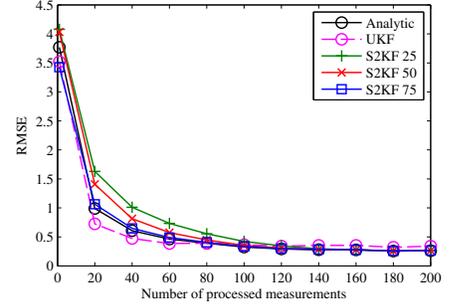
where \mathbf{w}_k denotes an additive noise term.



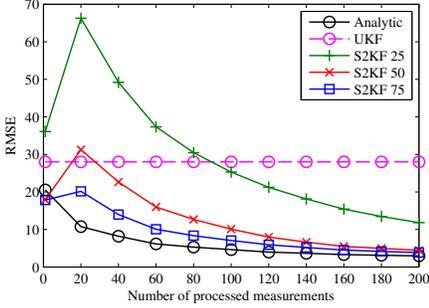
(a) Axis-aligned extended rectangular target with position \underline{c}_k , extent \underline{l}_k , and a target surface point \underline{z}_k .



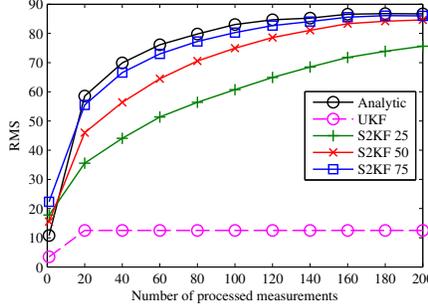
(b) Representative simulation run.



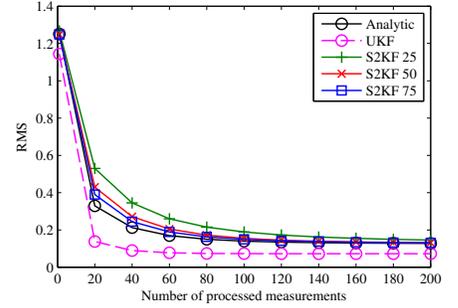
(c) RMSE for the target position.



(d) RMSE for the target area.



(e) RMS for the target region coverage in %.



(f) RMS for the target position variance.

Figure 4: Representative simulation run (b) with extended target (red line), randomly generated noisy measurements (magenta dots), analytic LRKF estimate (black dashed-line), 50 sample S^2KF estimate (blue dotted line) and UKF estimate (green dash-dotted-line). The UKF extent failure can be clearly seen. Evaluation results (c) – (f).

Unfortunately, as mentioned in [17], linear estimators, including the S^2KF as well, are unsuitable for tracking extended targets modeled as multiplicative noise. To overcome this issue, we seize the authors suggestion and extend the measurement equation (23) to match the best quadratic estimator according to

$$\underline{y}_k = \begin{bmatrix} \underline{m}_k \\ (\underline{m}_k)^2 \end{bmatrix} = \begin{bmatrix} \mathbf{H} \cdot \underline{l}_k + \underline{c}_k + \underline{w}_k \\ (\mathbf{H} \cdot \underline{l}_k + \underline{c}_k + \underline{w}_k)^2 \end{bmatrix} .$$

To keep things simple, this evaluation uses a static target with extent $\underline{l} = [4, 2]^T$ located at $\underline{c} = [3, 5]^T$. Thus, the temporal evolution of \underline{x}_k is modeled as random walk, i.e., employing the linear system equation

$$\underline{x}_k^p = \underline{x}_{k-1}^e + \underline{v} ,$$

where \underline{v} is an additive, zero-mean Gaussian noise term with covariance

$$\mathbf{C}^v = \text{diag}(10^{-4}, 10^{-4}, 10^{-3}, 10^{-3}) .$$

The initial system state is given by $\hat{\underline{x}}_0^e = [1, 1, 0, 0]^T$ and $\mathbf{C}_0^e = \mathbf{I}_4$. At each time step, we receive a single measurement from the target surface corrupted by additive, zero-mean Gaussian noise with unit covariance. Due to the fact that the S^2KF requires a measurement noise described in terms of a Gaussian distribution³, we approximate the uniformly

³This is a consequence of the S^2KF relying on explicit sampling a Gaussian distribution.

distributed multiplicative noise \mathbf{H} as Gaussian distribution by means of moment matching.

We compare the following estimators:

- Exact, analytic moment calculation (analytic LRKF) using [18] to serve as theoretical optimal estimation solution.
- The widely used UKF with equally weighted samples.
- Three S^2KF instances with 25, 50, and 75 samples, respectively, in order to demonstrate the convergence of the S^2KF towards the analytic LRKF.

Figure 4c depicts the root mean squared error (RMSE) for the estimated target position. It shows that all S^2KF instances are close to the analytic LRKF, and the S^2KF instance using 75 samples is nearly identical to it. Moreover, it shows the expected convergence of the S^2KF towards the analytic LRKF. At the beginning, the UKF is closest to the real target position, but after incorporating over 100 measurements, it becomes slightly worse than the other filters. Nevertheless, as the analytic LRKF defines the best possible LRKF estimate, the UKF in fact yields no better results than the S^2KF instances.

Figure 4d shows the RMSE for the estimated target area, i.e., the estimated target extent. The S^2KF convergence towards the analytic LRKF is even more evident and the 75 sample instance yields the best S^2KF results. As opposed to this, the UKF area estimate is clearly incorrect. More precisely, as the RMSE is constantly 28 m^2 , the initial axis lengths of 1 m were left unchanged, as the true target area amounts to 32 m^2 (see Figure 4b).

In order to assess the overall target estimate, we combine position and extent estimation by employing the intersection-over-union measure [19] according to

$$M := \frac{\int_{R_e \cap R_t} dx dy}{\int_{R_e \cup R_t} dx dy} \quad 0 \leq M \leq 1,$$

where R_e denotes the respective estimated and R_t the true target region in \mathbb{R}^2 .

Figure 4e displays the resulting root mean square (RMS) target region coverage. The results fit well into the already discussed position and area estimation errors. On the one hand, it shows the constantly wrong estimated 4 m² target region by the UKF, which lies completely within the true target region (see Figure 4b). This results in only a $\frac{4}{32} = 12.5\%$ region coverage. On the other hand, it is clearly visible that the analytic solution possesses the best region coverage, and that the employed S²KF instances continue their convergence towards the analytic LRKF. As with the estimation errors, the S²KF using 75 samples turns out to be the best of all employed sample-based LRKFs.

An interesting side-effect of the UKF's inability to track the target extent can be seen in Figure 4f. It depicts the RMS position variance, i.e., $\text{Var}\{c_k^x + c_k^y\}$, of each filter estimation. Whereas all S²KF instances and the analytic solution converge to the same RMS variance, the UKF converges to a much smaller value, which can be interpreted as being more confident of its estimated position, as it should be theoretically.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we introduced a new accurate LRKF called Smart Sampling Kalman Filter (S²KF). It is based on optimally LCD-generated Dirac Mixture approximations of standard normal distributions comprising an arbitrary number of samples, which are placed in the entire state space. Hence, the S²KF can be seen as the ultimate generalization of all sample-based LRKFs.

After describing sample generation and the actual filter procedures, we evaluated the S²KF by means of extended target tracking. The new filter showed the expected convergence of the S²KF towards the analytic LRKF and outperformed the widely used UKF. As the S²KF encompasses the same interface as the UKF, the S²KF can easily replace it in order to enhance existing and future filtering applications.

Although this paper focused on Gaussian noise and occurring uniform noise processes during the evaluation were reduced to Gaussian noise, the LCD approach can be used for optimal, deterministic sampling of other, non-Gaussian noise densities, and, hence, let the S²KF cope with non-Gaussian noise processes as well.

Future work will be done on adaptively determining how many samples are required per filter step in order to minimize the computational effort.

VII. ONLINE RESOURCES

An implementation of the new Smart Sampling Kalman Filter can be found online under [20]. Also the tracking simulation from Sec. V is available under [21].

REFERENCES

- [1] T. Lefebvre, H. Bruyninckx, and J. D. Schutter, "Kalman filters for non-linear systems: a comparison of performance," in *International Journal of Control*, vol. 77, no. 7, 2004, pp. 639–653.
- [2] —, "The linear regression kalman filter," in *Nonlinear Kalman Filtering for Force-Controlled Robot Tasks*, ser. Springer Tracts in Advanced Robotics. Springer, 2005, vol. 19.
- [3] R. E. Kalman, "A new approach to linear filtering and prediction problems," in *Transaction of the ASME - Journal of Basic Engineering*, March 1960, pp. 35–45.
- [4] M. F. Huber, F. Beutler, and U. D. Hanebeck, "Semi-analytic gaussian assumed density filter," in *Proceedings of the 2011 American Control Conference (ACC 2011)*, San Francisco, California, USA, June 2011.
- [5] S. J. Julier and J. K. Uhlmann, "A new extension of the kalman filter to nonlinear systems," in *11th Int. Symp. Aerospace/Defense Sensing, Simulation and Controls*, 1997, pp. 182–193.
- [6] —, "Unscented filtering and nonlinear estimation," in *Proceedings of the IEEE*, vol. 92, no. 3, March 2004, pp. 401–422.
- [7] R. Turner and C. E. Rasmussen, "Model based learning of sigma points in unscented kalman filtering," *Neurocomputing*, vol. 80, pp. 47–53, 2012.
- [8] J. Dunik, M. Šimandl, and O. Straka, "Unscented kalman filter: Aspects and adaptive setting of scaling parameter," in *IEEE Transactions on Automatic Control*, vol. 57, no. 2, September 2012, pp. 2411–2416.
- [9] M. F. Huber and U. D. Hanebeck, "Gaussian filter based on deterministic sampling for high quality nonlinear estimation," in *Proceedings of the 17th IFAC World Congress (IFAC 2008)*, vol. 17, no. 2, Seoul, Republic of Korea, July 2008.
- [10] J. Dunik, O. Straka, and M. Šimandl, "The development of a randomised unscented kalman filter," in *Proceedings of the 18th IFAC World Congress*, Milan, Italy, August 2011, pp. 8–13.
- [11] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York Chichester Weinheim Brisbane Singapore Toronto: Wiley-Interscience, 2001.
- [12] M. Roth and F. Gustafsson, "An efficient implementation of the second order extended kalman filter," in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, July 2011.
- [13] M. Nørgaard, N. K. Poulsen, and O. Ravn, "New developments in state estimation for nonlinear systems," *Automatica*, vol. 36, no. 11, pp. 1627–1638, 2000.
- [14] O. C. Schrempf and U. D. Hanebeck, "Dirac mixture approximation for nonlinear stochastic filtering," in *Informatics in Control, Automation and Robotics — Selected Papers from the International Conference on Informatics in Control, Automation and Robotics 2007*, ser. Lecture Notes in Electrical Engineering. Springer, September 2008, vol. 24, pp. 287–300.
- [15] U. D. Hanebeck, M. F. Huber, and V. Klumpp, "Dirac mixture approximation of multivariate gaussian densities," in *Proceedings of the 2009 IEEE Conference on Decision and Control (CDC 2009)*, Shanghai, China, December 2009.
- [16] I. Gilitschenski and U. D. Hanebeck, "Efficient deterministic dirac mixture approximation (to appear)," in *Proceedings of the 2013 American Control Conference (ACC 2013)*, Washington D. C., USA, June 2013.
- [17] M. Baum, F. Faion, and U. D. Hanebeck, "Modeling the target extent with multiplicative noise," in *Proceedings of the 15th International Conference on Information Fusion (Fusion 2012)*, Singapore, July 2012.
- [18] R. Kan, "From moments of sum to moments of product," in *Journal of Multivariate Analysis*, March 2008, vol. 99, no. 3, pp. 542–554.
- [19] B. Sapp, C. Jordan, and B. Taskar, "Adaptive pose priors for pictorial structures," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2010, pp. 422–429.
- [20] J. Steinbring, "Cloud runner – smart sampling kalman filter (s2kf)," May 2013. [Online]. Available: <http://www.cloudrunner.eu/algorithm/124/smart-sampling-kalman-filter-s2kf/>
- [21] —, "Cloud runner – tracking a rectangle extended target using lrkfs," May 2013. [Online]. Available: <http://www.cloudrunner.eu/algorithm/125/tracking-a-rectangle-extended-target-using-lrkfs/>