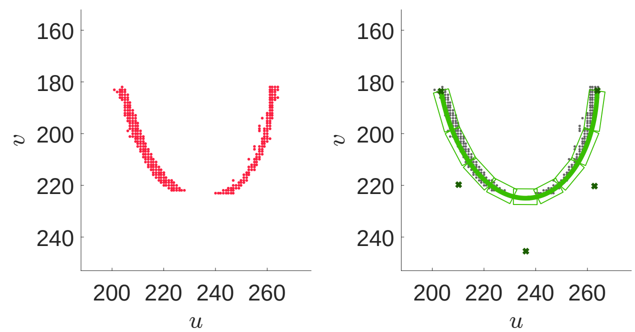
Tracking Elongated Extended Objects Using Splines

Antonio Zea, **Florian Faion**, and **Uwe D. Hanebeck**
Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
Institute for Anthropomatics and Robotics
Karlsruhe Institute of Technology (KIT), Germany
antonio.zea@kit.edu, florian.faion@kit.edu, uwe.hanebeck@ieee.org

***Abstract*—In this paper, we propose a novel approach to track elongated, curved extended targets by representing their shapes with splines. Elongated shapes are forms whose length is much larger than their width, and can be found in many places, such as in connected vehicles like trains, in group targets like a caravan moving along a curved street, or even when estimating the pose of a person. A particular property of these targets is that we cannot assume that their shape is rigid, as they can be expected to bend and deform as they move. This raises the need of continuously estimating their length, width, and curve characteristics as well as their position. We introduce a straightforward approach to track these shapes using splines, such as Bézier curves. By approximating these curves as rectangle chains, we can derive a simple closed-form likelihood function for use in a recursive Bayesian estimator. We also show that this approach can be easily extended to exploit negative measurements, i.e., clutter known not to stem from the target. This allows the estimator to be robust and maintain accuracy even in cases of low measurement quality. Finally, we evaluate the proposed approach using real data.**



(a) Measurements. (b) Bézier curve representation.

Figure 1: Left side, the partially occluded target being tracked (measurements in red). Right side, target representation using an open Bézier curve, approximated using rectangles. The crosses represent the control points.

I. Introduction

In this paper, we deal with tracking the parameters of an elongated, moving object. There are different approaches in literature to handle this task which depend on the information the sensor can provide, such as the quality and type of measurements. Classical target tracking techniques are concerned with the case where only a single source can be resolved, and the target can be assumed to be a single point. In this case, only the position and the motion parameters are of interest. Once multiple sources can be observed, robustness and accuracy are improved by also taking into account the target shape, and it may become necessary to simultaneously estimate the target extent and its orientation in addition to its position, leading to the field of extended object tracking (EOT). This raises the need for shape models, which associate incoming measurements to a given shape. The complexity of these models depends on the available information, building a continuum that ranges from simple but robust approximations, to detailed and accurate reconstructions.

Literature provides a multitude of approaches that deal with EOT. In case of low available information, the target is approximated as a simple shape such as lines [1], polynomials [2], or ellipses using random matrices [3]. As more information becomes available, the shape can become more detailed. One approach consists of constructing complex forms by combining multiple ellipses [4], [5], leading to the necessity of associating measurements to each ellipse. Other approaches parameterize the boundary radially from a central point, using Fourier series [6], extended Gaussian images [7], or Gaussian processes [8]. These models generally assume that the target is either convex or star-convex.

On a related line, tracking techniques can also be categorized based on the kinds of measurements they can process. The general approach is to exploit measurements that stem from the target, denoted as *positive* measurements. However, *negative* measurements, which are known not to stem from the object, are also valuable as they indicate where the target cannot possibly be, even if they are generally discarded as useless clutter. Exploiting negative measurements for tracking has been treated in [9]–[11]. Specific to EOT, the authors proposed a model to incorporate negative observations in [12].

In this paper, we focus on estimating elongated, moving target shapes such as the object visible in Fig. 1a. These shapes can appear, for example, when observing group objects such as a caravan of vehicles following a curved street, or when tracking connected targets such as trains and human limbs, or machinery with connected parts such as cranes. This presents the following challenges. First, the shape is generally neither convex nor star-convex. Second, we are dealing with an open curve, i.e., the start and end points do not meet. The width (or informally, the "thickness") and the length of the curve also need to be estimated. Third, while most approaches assume that the object is rigid, i.e., the shape may only move or be rotated, in this case we assume that the shape may also be bent, deformed, or compressed. Fourth, due to partial occlusions or sensor artifacts, we cannot assume that the entire shape will be visible in a single scan, or even that the visible parts will be connected.

The topic of elongated shapes has been discussed in the context of computer vision, for example in [13]–[15]. However, while the focus has been mainly in dealing with grid images, in the field of EOT measurements may arrive as unstructured points with individual uncertainties, especially in scenarios with multi-sensor fusion. Furthermore, in many cases, negative measurements are available, which can provide valuable information in cases of low measurement quality. In this paper, we propose a new approach for elongated shapes which takes into account these two issues. We will develop a shape model which represents the target shape as a Bézier curve of arbitrary complexity (Fig. 1b). By allowing its control points to move, we also permit the modeling of bends and compressions. Then, by approximating this curve as a chain of rectangles, we derive a simple likelihood function which can be used in a recursive Bayesian estimator. We also show that this approach can easily be extended to also incorporate negative measurements.

This paper is structured as follows. Sec. II illustrates the problem formulation, and Sec. III describes how extended objects are modeled. The proposed approach is introduced in Sec. IV, and details of the implementation are presented in Sec. V. An evaluation of the proposed ideas is described in Sec. VI, and Sec. VII concludes this paper.

II. Problem Formulation

The task explored in this paper is estimating the parameters of an extended target, in particular its shape and pose, based on incoming noisy point measurements from its surface. The state vector, containing all the required shape and pose parameters, is denoted as $\underline{x}$. The target shape is denoted by the set of points $\mathcal{Z}^x \subset \mathbb{R}^d$. The incoming measurements $\mathcal{Y} = \{\underline{y}_1, \cdots, \underline{y}_n\}$ are in Cartesian coordinates from $\mathbb{R}^d$. For simplicity, we will focus in this paper on the case where $d = 2$.

Each measurement $\underline{y}_i$ is assumed to have been generated by the following process. In the first step, a source point $\underline{z}_i \in \mathbb{R}^2$ is drawn from the target shape $\mathcal{Z}^x$. Then, in the second step, $\underline{z}_i$ is corrupted by additive sensor noise as modeled by

$$\underline{y}_i = \underline{z}_i + \underline{v}_i \ ,$$

yielding the measurement $\underline{y}_i$. The noise term $\underline{v}_i$ is assumed to be zero-mean Gaussian distributed with covariance matrix $\mathbf{R}_i$, assumed to have the form $\mathbf{R}_i = \sigma_{v,i}^2 \cdot \mathbf{I}$. The noise terms of different measurements are assumed to be independent from each other and from the state. As a consequence of unpredictable occlusions and sensor artifacts, we do not assume that the number of measurements carries information about the target.

Finally, measurements may be received at different discrete time steps, during which the state may also evolve according to a dynamic model. We denote the time step using the subindex k, i.e., the state at the time step k is $\underline{x}_k$ and the measurements at that point have the form $\underline{y}_{k,i}$.

III. Shape Models for Extended Objects

In this section, we explore how to model extended objects probabilistically, based on the generative model from Sec. II. The key idea is to develop a conditional density $\mathrm{p}(\mathcal{Y}_k \,|\, \underline{x}_k)$ which describes how the measurement set $\mathcal{Y}_k$ is associated to a given state $\underline{x}_k$. Then, by interpreting this term as a *likelihood* function for $\underline{x}_k$, we can derive a recursive Bayesian estimator for extended object targets. We exploit the fact that the noise terms are independent from each other to obtain

$$\mathrm{p}(\mathcal{Y}_k \,|\, \underline{x}_k) = \prod_{i=1}^{n} \mathrm{p}(\underline{y}_{k,i} \,|\, \underline{x}_k) \ .$$

The advantage of this result is that, in the following derivations, we only need to concern us with a single measurement $\underline{y}_{k,i}$ and process each term $\mathrm{p}(\underline{y}_{k,i} \,|\, \underline{x}_k)$ separately, as the results can be simply fused at the end by multiplying them. In the following, to improve legibility, we will drop the subindices i and k for sources and measurements unless needed.

A. Dealing with the Association Problem

The main challenge when deriving $\mathrm{p}(\underline{y} \,|\, \underline{x}_k)$ is that $\underline{y}$ cannot be directly associated to $\underline{x}_k$, as the source $\underline{z}$ that generated the measurement is generally unknown due to the sensor noise. This issue is known as the *association problem*. And in contrast with fields like multiple object tracking, it is difficult to enumerate a discrete list of hypotheses, as the number of potential sources is infinite.

A commonly used approach to deal with this issue, which we denote as *Greedy Association Model* [16], consists of simply associating the measurement to the "closest source", i.e., the source which minimizes some sort of metric such as the Euclidian distance. The task of the estimator, then, is to ensure the shape fits the incoming measurements the best way possible. However, for the open curves treated in this paper, this technique presents a challenge we denote as the *length problem*. This can be visualized in Fig. 2, where the measurements (gray) already have a minimal distance to the shape (cyan), and any other shape no matter how large will be equally likely as long as it also contains all the possible sources. Because of process noise, the length (and the width) will eventually diverge and become arbitrarily large, and the estimator has no way to recover. This presents a problem, as for this paper we are specifically interested in estimating the length and width parameters.

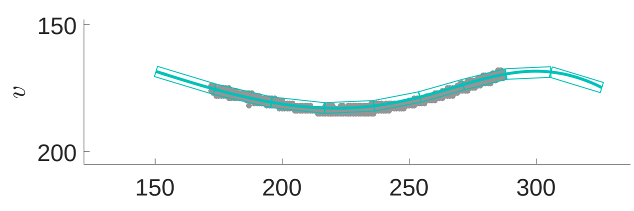


Figure 2: Example estimate using an association to the nearest source. The length and width cannot be estimated, and may become arbitrarily large. Measurements in gray, shape estimate in cyan.

B. Spatial Distribution Model

A method to address the length problem is by using *Spatial Distribution Models* (SDMs) [1]. The idea is to model how sources are selected from the shape $\mathcal{Z}_k^x$, e.g., during observation by the sensor, as the probability density $\mathrm{p}(\underline{z} \,|\, \underline{x}_k)$. Then, we obtain the likelihood function

$$\mathrm{p}(\underline{y} \,|\, \underline{x}_k) = \int_{\mathbb{R}^2} \mathcal{N}(\underline{y} - \underline{z}; \underline{0}, \mathbf{R}) \cdot \mathrm{p}(\underline{z} \,|\, \underline{x}_k) \; \mathrm{d}\underline{z} \ , \qquad (1)$$

and thus, we avoid an explicit association by implicitly associating $\underline{y}$ to every possible source in $\mathcal{Z}_k^x$.

Based on this idea, different techniques have been developed that make different assumptions on $\mathrm{p}(\underline{z} \,|\, \underline{x}_k)$, the shape being considered, and simplifications in order to make estimation more tractable [1], [3]–[6]. The main challenge is that, except for a few shapes, (1) is generally difficult to calculate in closed-form. Some works address this by treating $\mathrm{p}(\underline{z} \,|\, \underline{x}_k)$ as Gaussian distributed [3]. In this paper, however, we will assume that $\underline{z}$ is drawn uniformly from $\mathcal{Z}_k^x$, an idea also considered in [1], [17], [18]. We obtain

$$\mathrm{p}(\underline{z} \,|\, \underline{x}_k) = \frac{1}{|\mathcal{Z}_k^x|} \mathbf{1}_{\mathcal{Z}_k^x}(\underline{z}) \ ,$$

where $|\mathcal{Z}_k^x|$ is the shape area, and $\mathbf{1}_{\mathcal{Z}_k^x}(\cdot)$ is the indicator function of $\mathcal{Z}_k^x$. By plugging this into (1), we obtain

$$\begin{aligned} \mathrm{p}(\underline{y} \,|\, \underline{x}_k) &= \int\limits_{\mathbb{R}^2} \mathcal{N}(\underline{y} - \underline{z}; \underline{0}, \mathbf{R}) \cdot \frac{1}{|\mathcal{Z}_k^x|} \mathbf{1}_{\mathcal{Z}_k^x}(\underline{z}) \ \mathrm{d}\underline{z} \\ &= \frac{1}{|\mathcal{Z}_k^x|} \int\limits_{\mathcal{Z}_k^x} \mathcal{N}(\underline{y} - \underline{z}; \underline{0}, \mathbf{R}) \ \mathrm{d}\underline{z} \ . \end{aligned} \tag{2}$$

However, as mentioned before, in real-life scenarios it is usually impossible to obtain a reasonable approximation of $\mathrm{p}(\underline{z} \,|\, \underline{x}_k)$ consistent with the received measurements. An important case where this happens is in the case of unexpected occlusions or sensor artifacts, as seen in Fig. 3, which leads to estimation bias and low robustness.

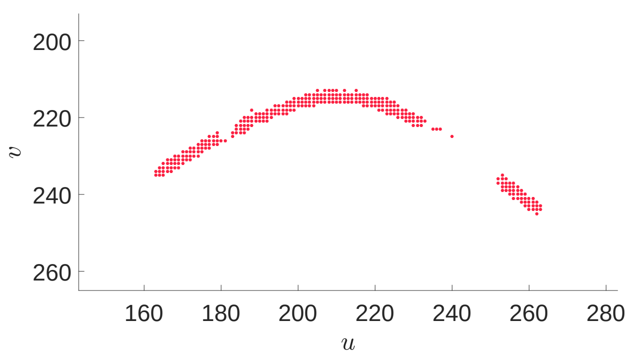


Figure 3: Example case of a partially occluded target, measurements in red. The source distribution cannot be known a priori, and the visible parts are not connected.

C. Exploiting Clutter for Extended Objects

In [12], the authors presented an alternative way to model extended objects, capable of incorporating "negative" measurements, i.e., a position in which the target *cannot* be, in addition to traditional positive measurements that stem from the target. Negative measurements, usually discarded in the segmentation process as unusable clutter, become invaluable in cases of low information, such as during initialization or in the presence of occlusions, when few or no positive measurements are available.

For this approach, we make the following assumptions about the sensor. First, in contrast to traditional scenarios, we assume the sensor produces a set of measurements $\mathcal{Y}$ in the same way as described in Sec. II, but the generating sources are not necessarily part of the target shape. Instead, each measurement takes the form of a pair $\left[\underline{y}^p, y^t\right]$, which consists of a position $\underline{y}^p \in \mathbb{R}^2$, and a corresponding type $y^t \in \{\odot^+, \odot^-\}$. A measurement of type $\odot^+$ is known to have been generated by the target, while a measurement of type $\odot^-$ is assured not to stem from it. Thus, the traditional models presented in Sec. III can be interpreted as special cases that only exploit measurements of type $\odot^+$. Second, we assume that the measured positions, independent of their type, are uniformly distributed in the region F visible by the sensor, i.e., its field of view. As motivation for this assumption, consider a rotating time-of-flight (TOF) sensor that samples its surroundings. The sensor will always sample the same angles each cycle, always producing a measurement for each $\underline{y}^p$, but only in combination with the corresponding type y^t can we gain information about the target. The difference is that, while traditional approaches would discard all measurements not produced by the target, we now aim to exploit all types of measurements.

Analogously to Sec. III-B, we will now derive a likelihood function $\mathrm{p}(\underline{y}^p, y^t \,|\, \underline{x}_k)$. First, as we mentioned before, $\underline{y}^p$ is assumed to be sampled uniformly from F, i.e.,

$$\mathrm{p}(\underline{y}^p \,|\, \underline{x}_k) = \frac{1}{|F|} := c_F \ .$$

For the sake of formality, we say that if $\underline{y}^p$ is outside of F, then $\mathrm{p}(\underline{y}^p \,|\, \underline{x}_k) = 0$. Second, we observe that the following equation holds,

$$\mathrm{p}(\underline{y}^p, \odot^+ \,|\, \underline{x}_k) + \mathrm{p}(\underline{y}^p, \odot^- \,|\, \underline{x}_k) = \mathrm{p}(\underline{y}^p \,|\, \underline{x}_k) \ . \tag{3}$$

This can be easily proven by interpreting it as a marginalization of y^t out of $\mathrm{p}(\underline{y}^p, y^t \,|\, \underline{x}_k)$. From these two equations, we obtain the following two results,

$$\begin{aligned} \mathrm{p}(\underline{y}^p, \odot^+ \,|\, \underline{x}_k) &= \mathrm{p}(\underline{y}^p \,|\, \underline{x}_k) \cdot \mathrm{p}(\odot^+ \,|\, \underline{y}^p, \underline{x}_k) \\ &= c_F \cdot \mathrm{p}(\odot^+ \,|\, \underline{y}^p, \underline{x}_k) \ , \end{aligned} \tag{4}$$

and, by exploiting (3),

$$\begin{aligned} \mathrm{p}(\underline{y}^p, \odot^- \,|\, \underline{x}_k) &= \mathrm{p}(\underline{y}^p \,|\, \underline{x}_k) - \mathrm{p}(\underline{y}^p, \odot^+ \,|\, \underline{x}_k) \\ &= c_F \cdot \left(1 - \mathrm{p}(\odot^+ \,|\, \underline{y}^p, \underline{x}_k)\right) \ . \end{aligned} \tag{5}$$

Thus, we see that, in order to describe $\mathrm{p}(\underline{y}^p, y^t \,|\, \underline{x}_k)$, we only need the term $\mathrm{p}(\odot^+ \,|\, \underline{y}^p, \underline{x}_k)$. For brevity, we define

$$\mathcal{L}_k(\underline{y}^p) := \mathrm{p}(\odot^+ \,|\, \underline{y}^p, \underline{x}_k) \ .$$

In [12] it was shown that $\mathcal{L}_k(\underline{y}^p)$ could be easily gained from a traditional SDM simply by multiplying (1) by the area $|\mathcal{Z}_k^x|$. We apply this idea on the uniform SDM from (2), yielding

$$\mathcal{L}_k(\underline{y}^p) = \int\limits_{\mathcal{Z}_k^x} \mathcal{N}(\underline{y}^p - \underline{z}; \underline{0}, \mathbf{R}) \ \mathrm{d}\underline{z} \ , \tag{6}$$

Plugging this into (4) and (5), we obtain

$$\begin{aligned} \mathrm{p}(\underline{y}^p, \odot^+ \,|\, \underline{x}_k) &= c_F \cdot \mathcal{L}_k(\underline{y}^p) \\ \mathrm{p}(\underline{y}^p, \odot^- \,|\, \underline{x}_k) &= c_F \cdot \left(1 - \mathcal{L}_k(\underline{y}^p)\right) \ , \end{aligned}$$

and $\mathrm{p}(\underline{y}^p, y^t \,|\, \underline{x}_k) = 0$ when $\underline{y}^p \notin F$. As c_F is a state-independent constant, it can generally be ignored during the estimation process. In the following, we denote this model as SDM-N.

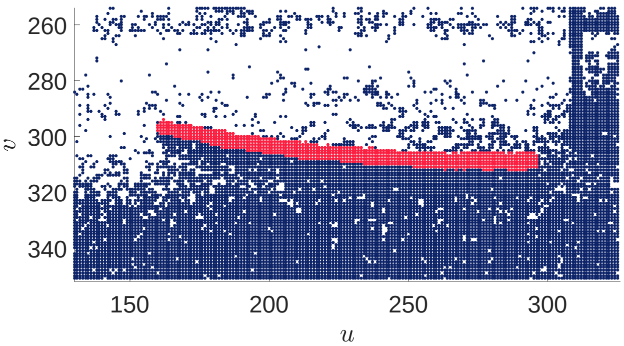


Figure 4: Example case where measurement positions are not uniform. Positive measurements in red, negative measurements in blue, empty spaces represent no information provided by the sensor due to material-related artifacts.

The advantage of this approach is its resilience against occlusions and incorrect assumptions of the source distribution. However, for real-life sensors, it cannot be assured that measurement positions are uniformly distributed in every circumstance. Fig. 4 shows such an example, where a sensor camera cannot observe the space above the target due to sensor artifacts, and the density of negative measurements is not uniform.

It becomes clear that both SDM and SDM-N bring their own advantages and disadvantages. The remainder of this paper will take into account both models, and the evaluation will explore the differences in more depth.

IV. Modeling Curves using Rectangle Chains

An important part of the task of tracking elongated, deformable targets is deciding which shape representation to be used. For this paper, we assume that target shape has the form of a "thick" Bézier curve, i.e., it has a non-zero width (Fig. 6a). The curve control points are either directly contained in $\underline{x}_k$, or are derived from its parameters. Furthermore, the number of control points can be arbitrary, but is known a priori. However, this representation raises the problem that the likelihood function may be too difficult to calculate. To solve this challenge, we do not work with the Bézier curve associated to a given $\underline{x}_k$ directly, but instead, we construct a rectangle chain approximation of it (Fig. 6b). Thus, the likelihood function becomes tractable due to the fact that, for rectangles, a simple closed-form solution can be found for the integrals in (2) and (6). The remainder of this section presents a general technique to calculate a rectangle chain approximation, and derives the SDM and SDM-N likelihood functions for this shape model. Then, Sec. V will describe the specific implementation used in the evaluation.

A. Building the Rectangle Chain

We now derive a simple way to approximate Bézier curves as a chain of connected rectangles, denoted as R_k. Fig. 5 presents a sketch of the procedure. First, we obtain a set of L control points $\underline{b}_k^1, \cdots, \underline{b}_k^L$, and a curve width w_k, from the state $\underline{x}_k$. Based on these control points, we can traverse all points in the Bézier curve using the function

$$\underline{b}_k^*(t) = \sum_{\ell=1}^{L} \binom{L}{\ell} t^\ell \, (1-t)^{L-\ell} \, \underline{b}_k^\ell \ ,$$

for $t \in [0,1]$. Second, we sample $m+1$ points $\underline{c}_k^0, \cdots, \underline{c}_k^m$, where

$$\underline{c}_k^j := \underline{b}_k^* \left(\frac{j}{m} \right) \ ,$$

as shown in Fig. 5a. Note that these sample points are distinct from the control points, and it does not need to hold that $m = L$. Finally, for each $1 \leq j \leq m$, we construct the rectangle R_k^j, which spans between $\underline{c}_k^{j-1}$ and $\underline{c}_k^j$, as described in Fig. 5b. We define the length $l_k^j := ||\underline{c}_k^j - \underline{c}_k^{j-1}||$, where $||\cdot||$ is the Euclidian norm. Furthermore, we define the spanning directions as illustrated in Fig. 5c, with $\underline{\alpha}_k^j := \left(\underline{c}_k^j - \underline{c}_k^{j-1} \right) / l_k^j$, and by rotating this vector by 90^o, we obtain $\underline{\beta}_k^j$.

B. Deriving a Likelihood Function

We can now derive the likelihood function $\mathrm{p}(\underline{y} \,|\, \underline{x}_k)$ for the SDM, in the form of

$$\begin{aligned} \mathrm{p}(\underline{y} \,|\, \underline{x}_k) &= \frac{1}{|\mathcal{Z}_k^x|} \int_{\mathbb{R}^2} \mathcal{N}(\underline{y} - \underline{z}; \underline{0}, \mathbf{R}) \cdot \mathbf{1}_{\mathcal{Z}_k^x}(\underline{z}) \, \mathrm{d}\underline{z} \\ &\approx \frac{1}{|R_k|} \int_{\mathbb{R}^2} \mathcal{N}(\underline{y} - \underline{z}; \underline{0}, \mathbf{R}) \left(\sum_{j=1}^{m} \mathbf{1}_{R_k^j}(\underline{z}) \right) \, \mathrm{d}\underline{z} \\ &= \frac{1}{|R_k|} \cdot \sum_{j=1}^{m} \underbrace{\int_{R_k^j} \mathcal{N}(\underline{y} - \underline{z}; \underline{0}, \mathbf{R}) \, \mathrm{d}\underline{z}}_{:= \mathcal{L}_k^j(\underline{y})} \ . \end{aligned}$$

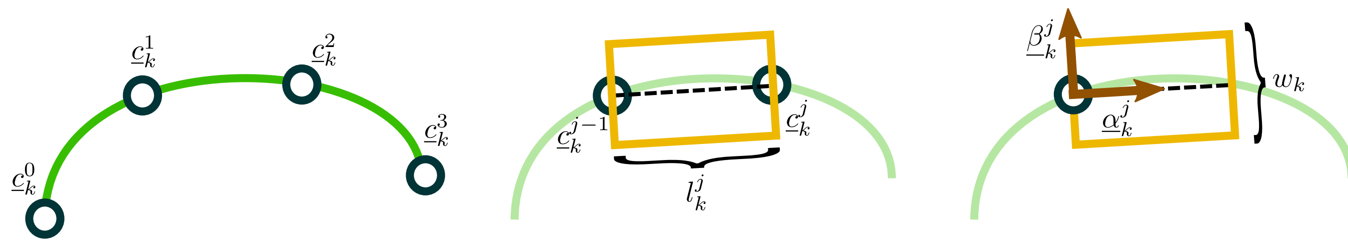
The closed-form solution of the integral $\mathcal{L}_k^j(\underline{y})$ can be found in the appendix. The term $|R_k|$ is the area of the entire rectangle chain, which easily follows from

$$|R_k| = \sum_{j=1}^{m} |R_k^j| = w_k \cdot \sum_{j=1}^{m} l_k^j \ .$$

The likelihood functions for the SDM-N follow directly as

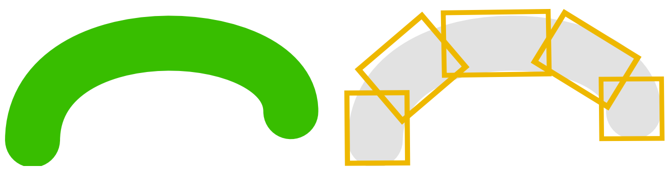
$$\begin{aligned} \mathrm{p}(\underline{y}^p, \odot^+ \,|\, \underline{x}_k) &= c_F \cdot \sum_{j=1}^{m} \mathcal{L}_k^j(\underline{y}^p) \\ \mathrm{p}(\underline{y}^p, \odot^- \,|\, \underline{x}_k) &= c_F \cdot \left(1 - \sum_{j=1}^{m} \mathcal{L}_k^j(\underline{y}^p) \right) \ . \end{aligned}$$

Some comments on the proposed rectangle chain approximation follow. First, when evaluating the likelihood function, all measurements are treated identically and independently from each other, and combining the results consists of a simple multiplication. This means that, during estimation, the complexity scales *linearly* in function of the number of measurements, the number of rectangles, and the number of state samples. This also allows for a very high degree of parallelization. Second, the number of rectangles can be



(a) Sample points. (b) Constructing the rectangle R_k^j. (c) Spanning directions of R_k^j.

Figure 5: A simple approach to construct a rectangle chain from a Bézier curve.



(a) Bézier curve. (b) Rectangle chain.

Figure 6: Representation of a Bézier curve as a chain of rectangles. Note that we assume that the curve is “thick”, i.e., it has a non-zero width w_k.

arbitrary, allowing for a flexible trade-off between accuracy and speed that can be changed at will. However, this approach has the following weakness. As can be seen in Fig. 6b, this simple construction does not prevent rectangles from overlapping on certain regions, and from leaving empty holes in others. Nonetheless, we will show that this construction still produces accurate results.

V. Implementation

In this section we present an example implementation of the proposed approach, which will be used for the evaluation in Sec. VI.

A. State Parametrization

The state vector $\underline{x}_k$ at the timestep k takes the form

$$\begin{aligned}\underline{x}_k^P &= \left[\underline{p}_k, s_k, w_k, \theta_k^1, \theta_k^2, \theta_k^3, \theta_k^4\right]^T \\ \underline{x}_k^V &= \left[\underline{v}_k, \omega_k^1, \omega_k^2, \omega_k^3, \omega_k^4\right]^T \\ \underline{x}_k &= \left[(\underline{x}_k^P)^T, (\underline{x}_k^V)^T\right]^T \quad ,\end{aligned}$$

leading to a 14-dimensional state. The vector $\underline{p}_k$ represents the center position, s_k the distance between control points, and w_k the curve width. The angles $\theta_k^1, \cdots, \theta_k^4$ serve to construct the 5 control points, in the form of

$$\begin{aligned}\underline{b}_k^3 &= \underline{p}_k \\ \underline{b}_k^2 &= \underline{b}_k^3 - s_k \cdot \left[\cos(\theta_k^2), \sin(\theta_k^2)\right]^T \\ \underline{b}_k^4 &= \underline{b}_k^3 + s_k \cdot \left[\cos(\theta_k^3), \sin(\theta_k^3)\right]^T \\ \underline{b}_k^1 &= \underline{b}_k^2 - s_k \cdot \left[\cos(\theta_k^1), \sin(\theta_k^1)\right]^T \\ \underline{b}_k^5 &= \underline{b}_k^4 + s_k \cdot \left[\cos(\theta_k^4), \sin(\theta_k^4)\right]^T \quad .\end{aligned}$$

These are the control points required for the chain construction described in Sec. IV-B. For the remaining terms, $\underline{v}_k$ represents the velocity of $\underline{p}_k$, and $\omega_k^1, \cdots, \omega_k^4$ the velocities of $\theta_k^1, \cdots, \theta_k^4$.

B. Initialization

When using a recursive estimator, it is very important to have an adequate initial estimate. The initialization is a difficult challenge for the non-convex shapes we are dealing with, given that the likelihood may have multiple incorrect modes. We propose the following approach using ideas from Principal Component Analysis. For the timestep $k = 0$, and given an initial set of measurements $\mathcal{Y}_0 = \{\underline{y}_{0,1}, \cdots, \underline{y}_{0,n}\}$, we calculate

$$\begin{aligned}\hat{\underline{y}}_0 &= \frac{1}{n}\sum_{i=1}^{n} \underline{y}_{0,i} \ , \text{ and} \\ \mathbf{C}_0^y &= \frac{1}{n}\sum_{i=1}^{n} \left(\underline{y}_{0,i} - \hat{\underline{y}}_0\right)\left(\underline{y}_{0,i} - \hat{\underline{y}}_0\right)^T \quad .\end{aligned}$$

Let the eigenvectors of $\mathbf{C}_0^y$ be $\underline{e}_1$ and $\underline{e}_2$, with corresponding eigenvalues λ_1 and λ_2, so that $\lambda_1 \leq \lambda_2$. Then, we define $\gamma := \text{atan2}\left(e_2^{(2)}, e_2^{(1)}\right)$ as the orientation of $\underline{e}_2$. Finally, we initialize $\underline{x}_0$ as

$$\begin{aligned}\underline{x}_0^P &= \left[\hat{\underline{y}}, 2 \cdot \lambda_2, 4 \cdot \lambda_1, \gamma, \gamma, \gamma, \gamma\right]^T \\ \underline{x}_0^V &= \left[\underline{0}, 0, 0, 0, 0\right]^T \\ \underline{x}_0 &= \left[(\underline{x}_0^P)^T, (\underline{x}_0^V)^T\right]^T \quad .\end{aligned}$$

Of course, this approach only produces satisfactory results if the initial shape is not too bent or coiled, such as Fig. 7. Otherwise, more elaborate initialization procedures may be necessary, which will be explored in future work.

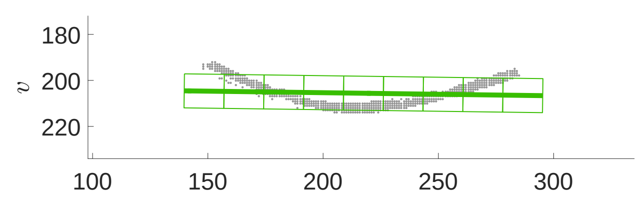


Figure 7: Example initialization in green, with the starting measurements in gray.

C. Dynamic Model

In this subsection, we will describe how the state $\underline{x}_k$ evolves in time for this implementation. Note that the SDM and the SDM-N presented previously are only concerned with deriving a likelihood function, and thus, they are not concerned with the dynamic model, and do not impose any constraint upon it. We will assume that $\underline{x}_k$ evolves according to a constant velocity model, i.e., at the timestep k and given a change in time Δt_k, the state evolution is described by

$$\underline{a}_k(\underline{x}_k) = \begin{bmatrix} \underline{p}_k + \Delta t_k \cdot \underline{v}_k \\ \theta_k^1 + \Delta t_k \cdot \omega_k^1 \\ \theta_k^2 + \Delta t_k \cdot \omega_k^2 \\ \theta_k^3 + \Delta t_k \cdot \omega_k^3 \\ \theta_k^4 + \Delta t_k \cdot \omega_k^4 \\ \vdots \end{bmatrix} .$$

The parameters not listed are assumed to remain unchanged.

D. Recursive Bayesian Estimator

As with the dynamic model, the proposed shape models do not impose any constraint upon the recursive Bayesian estimator being used, as long as they admit an explicit likelihood, such as the well-known particle filters. For this implementation, we will use a *Progressive Gaussian Filter* (PGF) [19], which represents the state uncertainty as a Gaussian distribution. Thus, at the timestep k, the state is represented by the probability density

$$f_k(\underline{x}_k) = \mathcal{N}(\underline{x}_k; \hat{\underline{x}}_k, \mathbf{P}_k) .$$

The estimation process consists of two steps,

1) the *update* step, where the filter corrects the knowledge about $\underline{x}_k$ based on the received measurements $\mathcal{Y}_k$ using Bayes' rule and the likelihood functions derived in Sec. IV-B,
2) and a *prediction* step, which lets the state uncertainty at the time step k evolve in time into the next step $k+1$, using the system equation

$$\underline{x}_{k+1} = \underline{a}_k(\underline{x}_k) + \underline{w}_k ,$$

where $\underline{a}_k(\cdot)$ is the function described in Sec. V-C, and $\underline{w}_k \sim \mathcal{N}(\underline{0}, \mathbf{Q}_k)$ represents a zero-mean Gaussian system noise term.

Note that, given that the state is Gaussian distributed and $\underline{a}_k(\cdot)$ is linear, the prediction step can be calculated analytically.

VI. Evaluation

In this section, we will evaluate the presented concepts using real sensor data from a Kinect v2 camera. For reference, the measurements shown in the previous pictures (such as Fig. 1) stem from these sensors. For the evaluation, the measurements together with the shape models will be in $\mathbb{R}^2$. Note that, while this device is capable of producing full three-dimensional data, for this evaluation we will only use the depth for segmentation. Extending the proposed models to $\mathbb{R}^3$ for use with depth sensors is a straightforward task, and part of our immediate future work.

A Kinect v2 sensor contains a color camera and a depth camera. It provides depth images (Fig. 8a), i.e., the intensity value of each pixel corresponds to a depth value, at a rate of 30 frames per second. By registering the color camera to the depth camera, we can associate a color to each pixel in the depth image. In order to find the measurements related to the target, we apply the following segmentation process. First, as we know a priori the color of the target object, we can mark as negative measurements all pixels that do not match this color, and as positive all measurements that do. Second, as we also know that the target is more than 4 m away from the camera, we mark as *invalid* all pixels behind this threshold (Fig. 8b). Note that we do not mark them as negative, as that would imply that the target is not at this position, and we cannot know this for sure a priori, as it may happen that the target is behind this occlusion. Finally, a bounding box around all positive measurements is calculated, inflated by 20 pixels in all directions. This box serves as the field of view, and all measurements outside of it are ignored (Fig. 8c).

The settings for the estimator follow. For the PGF, we employed $14 * 16 = 224$ state samples. For the measurement noise, we assume that $\mathbf{R}_{k,i} = 1\,\text{px}^2 \cdot \mathbf{I}$. While this may seem small, notice that the target is usually 4-7 pixels wide. For the process noise and the initial state covariance matrix, we define $\mathbf{Q}^b := \text{diag}(\mathbf{Q}^P, \mathbf{Q}^V)$, with $\mathbf{Q}^P = \text{diag}(1,1,1,1,1,10^{-4},10^{-4},10^{-4},10^{-4})$, and $\mathbf{Q}^V = \text{diag}(1,1,10^{-2},10^{-2},10^{-2},10^{-2})$. Then, we set $\mathbf{Q}_k = \mathbf{Q}^b$, and $\mathbf{P}_0 = 10 \cdot \mathbf{Q}^b$. Note that, as the pixel values are orders of magnitude larger than the angle values, their entries are correspondingly larger.

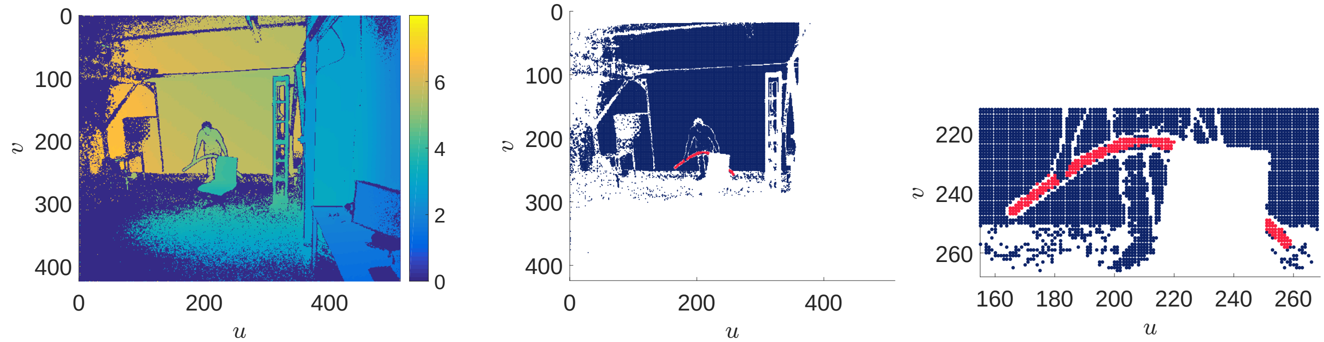
The models being evaluated are the SDM and the SDM-N. In order to measure the quality of the shape estimate, we require some sort of ground-truth, which is unavailable. Instead, we represent the real target shape as the set Y_k, constructed in the following way. For each positive measurement, we create a $1\,\text{px} \times 1\,\text{px}$ rectangle on that position, and then Y_k results as the union of all of these rectangles. Finally, we define the intersection over union function in the form of

$$I(k) := \frac{Y_k \cap R_k}{Y_k \cup R_k} ,$$

where R_k is the shape of the evaluated rectangle chain. Thus, $I(k) = 1$ means both shapes have a perfect match, and $I(k) = 0$ represents the exact opposite. Of course, $I(k)$ is not meant as an exact measure, and is instead more of a qualitative illustration. In the following, we describe the two evaluation scenarios, one with rapid motions and without occlusions, and one with slower motions but with occlusions.

In the first scenario, the target is moved and bent rapidly. Fig. 10 shows the results, where each timestep k represents $0.033\,\text{s}$, so that the entire run lasted a little more than 3 seconds. In this case, both models are capable of correctly estimating the length. However, the combination of the source probability distribution not being correct, together with the rapid motions, caused the width of the SDM to slowly collapse towards 0. The SDM-N, on the other hand, had less problems with estimating the width. This can be explained by the fact that the shape boundary contains several invalid measurements, which become larger the faster the object is moving. Both models interpret these gaps in a different way. On the one hand, an SDM treats a lack of measurements as evidence that the shape has shrunk. On the other hand, an SDM-N makes no assumptions about these gaps, and shrinks only in the presence of negative measurements. This difference becomes evident in Fig. 9a.

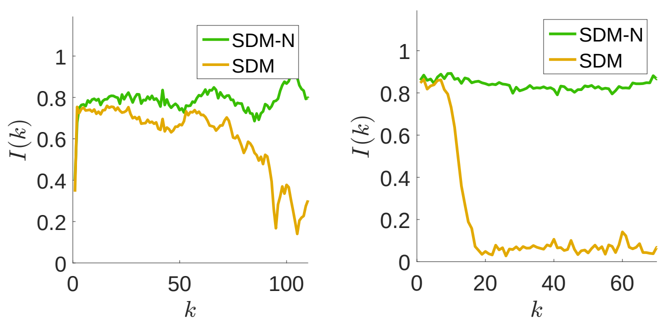
In the second scenario, the target is moving more slowly, but there is a significant occlusion between $k = 10$ and $k = 60$. Fig. 11 shows the results. Note that the measurements for $k = 53$, including the large occluded gap, can be seen more clearly in Fig. 8c, which was taken from this scenario. In this case, the inability for the SDM to estimate the width is more



(a) Depth image, distance in meters. (b) Positive and negative measurements. (c) Field of view.

Figure 8: Example setup for a target partially occluded by a chair. Positive measurements in red, negative in blue, gaps are invalid measurements. Note that the estimator needs to deal with some measurements being incorrectly classified.

egregious, as can be seen by the quick drop in Fig. 9b. However, the SDM-N still yields appropriate results, even considering that several measurements are misclassified, as can be seen in the left part of Fig. 8c. The SDM-N is also capable of reacting more quickly to changes in motion, due to the high availability of negative measurements. Nonetheless, ignoring the differences in the width estimation, both models are able to follow the length and deformations very closely.



(a) No occlusions. (b) With occlusions.

Figure 9: Intersection over union for the evaluated scenarios. A value of 1 is optimal, a value of 0 is a complete mismatch.

VII. Conclusion

In this paper, we focused on the task of tracking elongated extended objects that can be deformed or bent. The difficulty consisted of the fact that the target had the form of an open curve, i.e., the start and end points did not meet, making the estimation of its width and length more difficult. We proposed an approach to represent the target shape in the form of a Bézier curve, and during the evaluation of the likelihood function, we approximated this curve as a chain of rectangles. This allowed us to employ both a traditional SDM, and an extension called SDM-N, which can also exploit information from measurements that do not belong to the target. We evaluated the proposed approach using real sensor data. It turned out that both models were capable of estimating the length of the target, and reacted well to bends and deformations. However, the low quality of the sensor data prevented the traditional SDM from estimating the width correctly, while the SDM-N, which relies on the more numerous negative measurements, had few problems in this task.

Appendix

In this appendix, we will derive a closed-form solution to the integral

$$\mathcal{L}_k^j(\underline{y}) := \int_{R_k^j} \mathcal{N}(\underline{y} - \underline{z}; \underline{0}, \mathbf{R}) \, \mathrm{d}\underline{z} \ ,$$

related to the rectangle R_k^j. As a reminder, it holds that $\mathbf{R} = \sigma_v^2 \cdot \mathbf{I}$. First, we construct the 2×2 matrix

$$\mathbf{M}_k^j := \begin{bmatrix} \underline{\alpha}_k^j & \underline{\beta}_k^j \end{bmatrix} \ ,$$

that has the spanning directions of R_k^j as columns. Second, we calculate the following term

$$\underline{\eta}_k^j = \left[\eta_{k,1}^j, \eta_{k,2}^j\right]^T := \left(\mathbf{M}_k^j\right)^{-1} \cdot (\underline{y} - \underline{c}_k^{j-1}) \ .$$

Then, using the helper function

$$G\left(a_1, a_2, \mu, \sigma_v\right) := \frac{1}{2}\left(\operatorname{erf}\left(\frac{a_2 - \mu}{\sqrt{2}\,\sigma_v}\right) - \operatorname{erf}\left(\frac{a_1 - \mu}{\sqrt{2}\,\sigma_v}\right)\right)$$

we obtain the solution in the form of

$$\mathcal{L}_k^j(\underline{y}) = G\left(0, l_k^j, \eta_{k,1}^j, \sigma_v\right) \cdot G\left(-\frac{w_k}{2}, \frac{w_k}{2}, \eta_{k,2}^j, \sigma_v\right) \ ,$$

where $\operatorname{erf}(\cdot)$ is the error function, available in every modern statistics library.

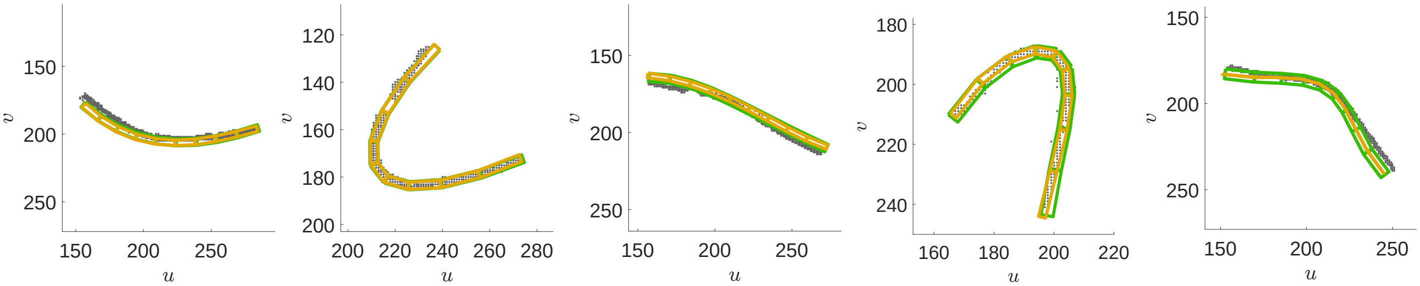
References

[1] M. Werman and D. Keren, “A Bayesian method for fitting parametric and nonparametric models to noisy data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 5, pp. 528–534, 2001.

[2] C. Lundquist, U. Orguner, F. Gustafsson, and S. Member, “Extended target tracking using polynomials with applications to road-map estimation,” *IEEE Transactions on Signal Processing*, pp. 1–12, 2011.

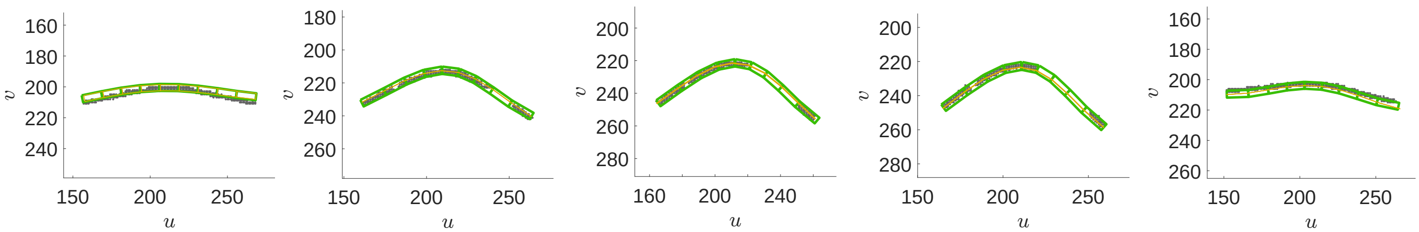
[3] M. Feldmann, D. Fränken, and W. Koch, “Tracking of extended objects and group targets using random matrices,” *Signal Processing, IEEE Transactions on*, vol. 59, no. 4, pp. 1409–1420, 2011.

[4] J. Lan and X. R. Li, “Tracking of maneuvering non-ellipsoidal extended object or target group using random matrix,” *IEEE Transactions on Signal Processing*, vol. 62, no. 9, pp. 2450–2463, May 2014.



(a) $k = 4$. (b) $k = 30$. (c) $k = 55$. (d) $k = 80$. (e) $k = 102$.

Figure 10: Scenario without occlusions. Measurements in gray, SDM in yellow, SDM-N in green.



(a) $k = 4$. (b) $k = 18$. (c) $k = 35$. (d) $k = 53$. (e) $k = 68$.

Figure 11: Scenario with occlusions. Measurements in gray, SDM in yellow (reduced to a simple line), SDM-N in green.

[5] K. Granstrom, P. Willett, and Y. Bar-Shalom, "An extended target tracking model with multiple random matrices and unified kinematics," in *Information Fusion (Fusion), 2015 18th International Conference on*, July 2015, pp. 1007–1014.

[6] M. Baum and U. D. Hanebeck, "Shape Tracking of Extended Objects and Group Targets with Star-Convex RHMs," in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, Chicago, Illinois, USA, 2011.

[7] L. Sun, X. Li, and J. Lan, "Extended object tracking based on support functions and extended gaussian images," in *Information Fusion (FUSION), 2013 16th International Conference on*, July 2013, pp. 1526–1533.

[8] N. Wahlstrom and E. Ozkan, "Extended target tracking using gaussian processes," *Signal Processing, IEEE Transactions on*, vol. 63, no. 16, pp. 4165–4178, 2015.

[9] W. Koch, "On exploiting 'negative' sensor evidence for target tracking and sensor data fusion," *Information Fusion*, vol. 8, no. 1, pp. 28–39, 2007.

[10] J. Hoffman, M. Spranger, D. Gohring, and M. Jungel, "Making use of what you don't see: negative information in markov localization," in *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, Aug 2005, pp. 2947–2952.

[11] W. Koch, "On 'negative' information in tracking and sensor data fusion: Discussion of selected examples," in *Proceedings of the Seventh International Conference on Information Fusion*, vol. 1. IEEE Publ. Piscataway, NJ, 2004, pp. 91–98.

[12] A. Zea, F. Faion, and U. D. Hanebeck, "Exploiting Clutter: Negative Information for Enhanced Extended Object Tracking," in *Proceedings of the 18th International Conference on Information Fusion (Fusion 2015)*, Washington D. C., USA, Jul. 2015.

[13] M. Schaap, I. Smal, C. Metz, T. van Walsum, and W. Niessen, *Information Processing in Medical Imaging: 20th International Conference, IPMI 2007, Kerkrade, The Netherlands, July 2-6, 2007. Proceedings*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, ch. Bayesian Tracking of Elongated Structures in 3D Images, pp. 74–85.

[14] D. Wesierski and P. Horain, "Pose-configurable generic tracking of elongated objects," in *2013 IEEE International Conference on Computer Vision*, Dec 2013, pp. 2920–2927.

[15] R. Toledo, X. Orriols, X. Binefa, P. Radeva, J. Vitria, and J. J. Villanueva, "Tracking elongated structures using statistical snakes," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 1, 2000, pp. 157–162 vol.1.

[16] F. Faion, A. Zea, and U. D. Hanebeck, "Reducing Bias in Bayesian Shape Estimation," in *Proceedings of the 17th International Conference on Information Fusion (Fusion 2014)*, Salamanca, Spain, Jul. 2014.

[17] N. Petrov, L. Mihaylova, A. Gning, and D. Angelova, "A Novel Sequential Monte Carlo Approach for Extended Object Tracking Based on Border Parameterisation," in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, Chicago, Illinois, USA, Jul. 2011.

[18] K. Gilholm and D. Salmond, "Spatial distribution model for tracking extended objects," in *Radar, Sonar and Navigation, IEE Proceedings-*, vol. 152, no. 5. IET, 2005, pp. 364–371.

[19] J. Steinbring and U. D. Hanebeck, "Progressive Gaussian Filtering Using Explicit Likelihoods," in *Proceedings of the 17th International Conference on Information Fusion (Fusion 2014)*, Salamanca, Spain, Jul. 2014.