Tracking Connected Objects Using Interacting Shape Models

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*—As sensor resolution increases, estimators tracking extended objects benefit from being able to closely model the shape of the target. However, as more shape details are incorporated, this usually leads to increasingly complex estimators. A more useful approach is to describe these shapes as a combination of simpler shapes connected to each other. In this paper, we propose a modular approach to estimate these combined targets in function of their simpler components. This allows the characteristics of each component to be encapsulated, and permits the combination of multiple filtering techniques as required by each component shape. This approach can be applied to track combined objects in a large variety of environments, such as excavators, robotic arms, wagon trains, and many others.**

***Keywords*—Extended object tracking, connected shapes, shape models, switching models.**

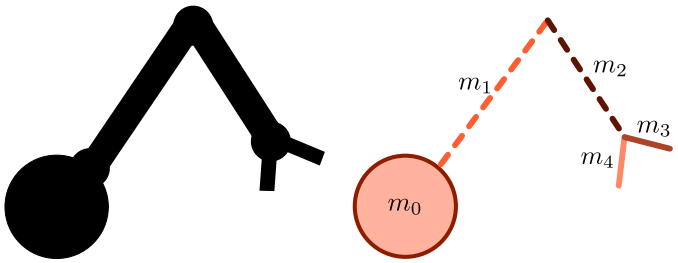
I. Introduction

The classic approach to track an object is to assume that measurements can only be generated from a single source. However, the increasing resolution of modern sensors allows the resolution of measurements that originate from different points on the target's boundary. In these cases, incorporating information about the shape of the target yields more robust and accurate results. This approach is denoted as extended object tracking.

Several techniques have been developed in literature to track simple objects based on noisy point measurements, in particular conic approximations [1]–[5] and sticks [6], [7], ideal for cases with very high noise. For conditions with better measurement quality, more details of the target's shape can be incorporated. This leads to approaches using splines [8], star-convex [9], and polygonal [10] approximations. However, as more information about the shape is incorporated, the estimators tend to become increasingly complex.

In this paper, we explore the approach of estimating complex shapes as the *combination* of simple shapes. In this way, a complex shape such as Fig. 1a can be subdivided into different *components*, see Fig. 1b, which reuse the same state parameters but interpret them individually. The estimated shape is then obtained from the combination, or mixture, of the individual component estimates. This gives the approach a measure of *modularity*, where the characteristics of each shape are encapsulated in their component, allowing for a combination of different shape models and filtering techniques.

The proposed approach can be seen as the intersection of two branches of object tracking. On the one hand, the use of multiple components makes the approach related to multiple object tracking [11]–[13]. On the other hand, the fact that the combined shape changes its behaviour depending on its components makes the approach related to dynamic multiple model estimation [14]–[16].



(a) Target being tracked. (b) Combined shape model.

Figure 1: Modeling a target as a combination of components.

This paper is structured as follows. Sec. II presents the problem formulation. Sec. III describes commonly used shape models. Sec. IV explains the theoretical aspects of combined shape models and Sec. V shows an implementation based on nonlinear Kalman filters. Then, Sec. VI presents an evaluation of the proposed approach. Finally, Sec. VII concludes the paper.

II. Problem Formulation

This paper deals with shape estimation based on received noisy point measurements. The parameters of the shape are contained in the state $\underline{x}$. In general, estimators treat the state as a random vector $\underline{\boldsymbol{x}}$, where the probability density function $p(\underline{x})$ represents the uncertain knowledge of the parameters. The received point measurements $\mathcal{Y} = \{\underline{y}_0, \cdots, \underline{y}_l\}$ are described in Cartesian coordinates.

Each measurement $\underline{y} \in \mathcal{Y}$ is related to the state $\underline{\boldsymbol{x}}$ through the *measurement model*, which can be divided into the *sensor model* and the *shape model*. The sensor model assumes that $\underline{y}$ originates from a measurement source $\underline{\tilde{z}}$ drawn from the target shape $\mathcal{Z}(\underline{x})$, and then is disturbed during observation by an additive noise term $\underline{v}$, as determined by

$$\underline{y} = \underline{\tilde{z}} + \underline{v} \ , \tag{1}$$

where $\underline{v}$ is drawn from the random variable $\underline{\boldsymbol{v}}$, which is Gaussian distributed with zero-mean and covariance matrix $\mathbf{C}^v$, i.e., $\underline{\boldsymbol{v}} \sim \mathcal{N}(\underline{0}, \mathbf{C}^v)$. The shape model describes how $\underline{\tilde{z}}$ is generated in function of $\underline{x}$, and depends on the characteristics of the

target shape, as described in Sec. III. Some shape models may additionally require a shape noise parameter, denoted as $\boldsymbol{s}$.

The measurement model can be described by the *measurement equation*

$$\underline{h}(\underline{x}, \underline{y}, \underline{v}, s) = \underline{0} \ , \tag{2}$$

which implicitly describes the shape and sensor models. The term $\underline{h}(\cdot, \cdot, \cdot, \cdot)$ is denoted as the *measurement function*. The relation between a given measurement $\underline{y}$ and the state $\underline{x}$ can be described probabilistically using the likelihood $p(\underline{y} \,|\, \underline{x})$, which is derived from (2). It is assumed that measurement sources and the noise parameters are drawn independently from each other and from the state.

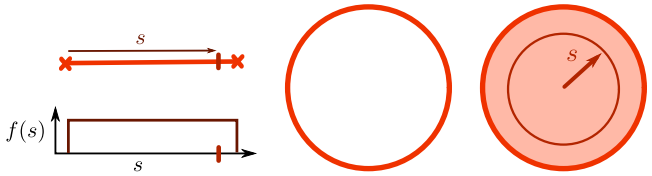
The state may also evolve in time. Thus, the state at time step k is denoted as $\underline{\boldsymbol{x}}_k$. This evolution is determined by the *system model* which can be described by the system function

$$\underline{\boldsymbol{x}}_{k+1} = \underline{a}(\underline{\boldsymbol{x}}_k, \underline{\boldsymbol{w}}_k) \ , \tag{3}$$

where $\underline{\boldsymbol{w}}_k$ is a random variable representing the *process noise*. The time index k will be dropped unless needed.

III. Modeling an Extended Shape

This section provides a short overview of commonly used models to describe extended shapes.



(a) Stick spatial distribution. (b) Circle GAM. (c) Disc RHM.

Figure 2: Common shape models.

A. Spatial Distribution

A *spatial distribution* [17] describes an extended object in the form of a set of point sources, where each source is assigned an explicit probability of generating a measurement. Thus, the shape $\mathcal{Z}(\underline{x})$ is described as

$$\mathcal{Z}(\underline{x}) = \{ \ \underline{\tilde{z}}(s, \underline{x}) \,|\, s \in \mathcal{S} \ \} \ ,$$

where $s \in \mathcal{S}$ serves to index each measurement source. In addition, each index s has a probability $p(s)$ that its corresponding source generates a measurement. Using this concept, we can visualize the source indices as being drawn from the random variable $\boldsymbol{s}$.

This leads to the measurement equation

$$\underline{h}(\underline{x}, \underline{y}, \underline{v}, s) = \underline{\tilde{z}}(s, \underline{x}) + \underline{v} - \underline{y} = \underline{0},$$

where the random variable $\boldsymbol{s}$ is treated as a shape noise parameter.

As an example, Fig. 2a shows a line segment in light red, where the scalar index s parameterizes the measurement sources, and $\boldsymbol{s}$ assigns each source a uniform probability.

B. Greedy Association Models

In many tracking scenarios, only a set $\mathcal{Z}(\underline{x})$ of measurement sources is given, but no assumption is made about the probability of individual points to generate measurements. Instead, only an implicit measurement equation is given, in the form of

$$\underline{h}(\underline{x}, \underline{y}, \underline{v}) = \underline{0} \ ,$$

where $\underline{h}(\cdot, \cdot, \cdot)$ is related to the pseudo-measurement $\underline{0}$. The measurement function $\underline{h}(\cdot, \cdot, \cdot)$ is generally related to some form of distance to a closest point, which is to be minimized. For this reason, we denote these models as *Greedy Association Models* (GAMs).

GAMs are generally employed for *curve fitting* [18]. Fig. 2b shows an example of a circle shape, characterized by the Euclidian distance to its center. On the one hand, this brings the advantage that there is no need to know how sources are distributed in $\mathcal{Z}(\underline{x})$, which is particularly useful in scenarios with occlusion. On the other hand, GAMs usually require an explicit consideration of the transformation of $\underline{v}$, which may lead to estimation *bias* [1], [2].

C. Random Hypersurface Models

A *Random Hypersurface Model* (RHM) [19] is an approach that combines spatial distributions and greedy association models. The idea is to describe the target shape based on a *shape family* $\mathcal{Z}_s(\underline{x})$, indexed with the parameter $s \in \mathcal{S}$. In addition, each index s is assigned a probability $p(s)$, indicating how likely it is that a measurement source is generated from $\mathcal{Z}_s(\underline{x})$. Using this concept, we can visualize the shape indices as being drawn from the random variable $\boldsymbol{s}$.

Each shape $\mathcal{Z}_s(\underline{x})$ is described as a GAM with measurement function $\underline{h}_s(\underline{x}, \underline{y}, \underline{v})$. This leads to the measurement equation

$$\underline{h}(\underline{x}, \underline{y}, \underline{v}, s) = \underline{h}_s(\underline{x}, \underline{y}, \underline{v}) \ ,$$

where $\boldsymbol{s}$ is treated as a shape noise parameter.

As an example, an RHM to describe a disc of radius r can be represented as follows. First, each subshape $\mathcal{Z}_s(\underline{x})$ is modeled as a circle of radius s, as seen in Fig. 2b. Finally, we assume that s is drawn from the interval $[0, r]$, in a similar way to Fig. 2a. It can be seen that both models together yield a disc such as Fig. 2c.

IV. Combining Shape Models

The objective of this paper is to model a single extended object, such as Fig. 1a, as a combination of multiple simple shapes, as seen in Fig. 1b.

A *combined shape model* is a set of shape models $\mathcal{M}$, where each *component* model $m \in \mathcal{M}$ has a probability $p(m)$ of generating a measurement source. Thus, the component models can be seen as being drawn from the random variable $\boldsymbol{m}$. The generative model for measurement sources can be described as, first, drawing a component m from $\boldsymbol{m}$, and then selecting a source from the corresponding component shape. $p(m)$ depends on several factors such as occlusion, sensor to object geometry, artifacts, and others, which may be difficult to model. Because of these issues, in many cases $p(m)$ cannot be known exactly.

The parameters of all components are contained in the state vector $\underline{x}$. Each component m has a known measurement model (2), described by $\underline{h}^m(\underline{x}, \underline{y}, \underline{v}, s^m)$, and a system model (3), in the form of $\underline{a}^m(\underline{x}, \underline{w}^m)$. All components use the same parameter state $\underline{x}$, but interpret it according to their individual models.

Each measurement $\underline{y}$ is modeled as having been generated by a unique component model. Thus, consecutive measurements being generated by different components can be interpreted as the result of *component model switching*. In addition, while each received measurement $\underline{y}$ is assumed to originate from $\mathcal{M}$, it is unknown which particular component generated it. However, backward inference and the incorporation of measurements requires each measurement to be *associated* to a given component. In consequence, a solution for estimating the parameters of a combined model can be found by drawing from the fields of multiple object tracking and dynamic multiple model estimation.

A. Combined Measurement Model

The measurement model for the component m is characterized by the likelihood $p(\underline{y} \,|\, m, \underline{x})$. As measurement sources and noise parameters are assumed to be independent, given a set of multiple measurements $\mathcal{Y}$, the component can treat multiple measurements separately as

$$p(\mathcal{Y} \,|\, m, \underline{x}) = \prod_{\underline{y} \in \mathcal{Y}} p(\underline{y} \,|\, m, \underline{x}) \ . \tag{4}$$

For the entire combined shape model, a measurement model for a single measurement can be derived using the law of total probability, in the form of

$$p(\underline{y} \,|\, \underline{x}) = \sum_{m \in \mathcal{M}} p(\underline{y} \,|\, m, \underline{x}) \cdot p(m \,|\, \underline{y}) \ , \tag{5}$$

where $p(m \,|\, \underline{y})$ determines the *association model* of $\underline{y}$ to m, described in Sec. IV-C.

However, it may hold that the probability for drawing a given m from $\boldsymbol{m}$ may be correlated with the components drawn for other measurements. In this case, the measurement models cannot be as easily separated as in (4), yielding instead

$$p(\mathcal{Y} \,|\, \underline{x}) = \prod_{\underline{y} \in \mathcal{Y}} \sum_{M \in \mathcal{M}^{|\mathcal{Y}|}} p(\underline{y} \,|\, M, \underline{x}) \cdot p(M \,|\, \mathcal{Y}) \ , \tag{6}$$

where M iterates over all possible *component chains* of length $|\mathcal{Y}|$. This raises a problem of hypothesis management, in particular given that the number of measurements can grow in time. One approach to solve this is the *Multiple Hypothesis Tracker* (MHT) [14], where (6) is approximated by maintaining a set of the N most likely component chains.

More tractable approaches can be derived from further simplifications depending on the component switch models.

B. Component Switch Models

In this paper, we will focus on two component switch models. For *random switch models*, the component probabilities are considered as independent. This allows us to treat each measurement independently, in a similar fashion to (4), in the form

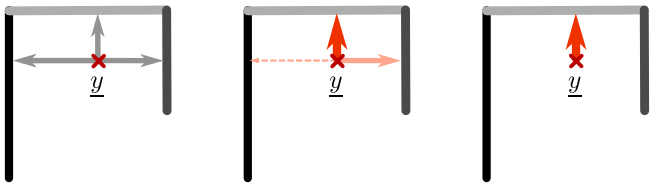
$$p(\mathcal{Y} \,|\, \underline{x}) = \prod_{\underline{y} \in \mathcal{Y}} p(\underline{y} \,|\, \underline{x}) \ .$$

These models can be applied in case of high state uncertainty and measurement noise.

For *Markov switch models*, the probability of a component depends only on the last component that generated a measurement. These models are governed by a fixed, known *transition matrix* $\mathbf{R}$, where each coefficient $r^{m,n}$ determines how probable it is that component m generates a measurement source right after component n did. As shown in (6), an exact solution in this case is intractable, but several methods to approximate a solution have been explored in literature [15], [16]. An application for these models is in high resolution sensors, where it is likely that a given component produces multiple, consecutive measurements.

C. Association Models

Given a measurement $\underline{y}$ and a component m, there are several approaches to describe the association model $p(m \,|\, \underline{y})$.



(a) Simple. (b) Weighted. (c) Direct.

Figure 3: Association models.

The simplest association model, visualized in Fig. 3a, is to assume

$$p(m \,|\, \underline{y}) = p(m),$$

i.e., that the probability that a given component generates a source is independent of the measurement. This approach is analogous to that used by spatial distributions in Sec. III-A. However, this form cannot appropriately deal with greedy association models. In addition, it also requires $p(m)$ to be known exactly, which in many cases might not be possible.

A different approach which does not depend as strongly on $p(m)$, is by applying Bayes' law, leading to

$$p(m \,|\, \underline{y}) \propto p(\underline{y} \,|\, m) \cdot p(m) \ , \tag{7}$$

where $p(\underline{y} \,|\, m)$ denotes the likelihood that $\underline{y}$ was generated by the component m. We denote this as *weighted association*, visualized in Fig. 3b. This approach has the advantage that an incorrect assumption of $p(m)$ can still lead to good estimation results, while also allowing for a consideration of all components for each measurement, which is useful in scenarios with high noise.

However, for cases of low measurement noise and state uncertainty, usually only a single component has a high value for $p(\underline{y} \,|\, m)$. In these cases, a simpler approach, which we denote as the *direct association* (Fig. 3c), is to only use the

most likely component m^L in function of $\underline{y}$. Thus, we select m^L in the form of

$$m^L := \arg\max_{m \in \mathcal{M}} \; p(\underline{y} \,|\, m) \cdot p(m) \;.$$

This leads to

$$p(m \,|\, \underline{y}) = \begin{cases} 1 & \text{if } m = m^L \\ 0 & \text{otherwise.} \end{cases}$$

The advantage is that m^L, and not explicitly $p(\underline{y} \,|\, m)$, is needed, allowing for less computationally expensive methods to be used which can yield an appropriate approximation of m^L. For instance, m^L can be chosen simply as the component containing the point closest to $\underline{y}$ given a suitable metric. This approach of a unique projection is commonly used in the literature, especially in works based on articulations with ICP [20], such as [21]. A combination of the last two approaches is also possible, i.e., by only considering a small subset of $\mathcal{M}$ around m^L.

D. Modeling Constraints

The method proposed in this paper is comparable to approaches for multiple object tracking, such as [11]–[13]. The key difference is that, in this paper, components are assumed to be *connected*, and thus, their parameters are highly correlated and cannot be assumed to be independent. This connection also imposes constraints between components shape parameters.

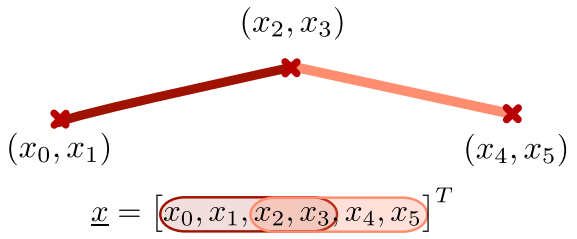


Figure 4: Constraints through parameter sharing.

In this paper, we model this connection in the parameter space, i.e., on the level on the state $\underline{x}$, instead of in the resulting shapes. As mentioned before, all components use the same parameters $\underline{x}$, they simply interpret the parameters individually. Thus, two components are assumed to be connected if they *share*, or reuse, a given parameter in their measurement equations. The resulting shape can be interpreted as an articulation, with the difference that two components could be modeled as connected even if their shapes do not intersect at any point. This approach is visualized in Fig. 4.

E. Estimator

The most widely used technique to estimate the parameters of the target shape is using a *maximum likelihood* estimator [22], [23]. Thus, using (4), the estimated $\underline{x}^{ml}$ in function of the set of measurements $\mathcal{Y}$ is obtained as

$$\underline{x}^{ml} = \arg\max_{\underline{x}} \; p(\mathcal{Y} \,|\, \underline{x}) \;.$$

Using Bayes' law, we can utilize prior knowledge $p(\underline{x})$ about the target and incorporate the new information from $\mathcal{Y}$, in the form of

$$p(\underline{x} \,|\, \mathcal{Y}) \propto p(\mathcal{Y} \,|\, \underline{x}) \cdot p(\underline{x}) \;. \tag{8}$$

From this, a *maximum a posteriori* estimate [23] can be derived as

$$\underline{x}^{map} = \arg\max_{\underline{x}} \; p(\underline{x} \,|\, \mathcal{Y}) \;.$$

However, both of these techniques require that all used measurements are known beforehand.

Of interest for this paper are *recursive Bayesian* estimators, which estimate the state in successive time steps, incorporating measurements as they arrive, while also allowing the state to evolve in time. Thus, the state at the time step k is denoted as $\underline{x}_k$, and the set of all incorporated measurements at that moment is $\mathcal{Y}_{1:k}$.

A recursive Bayesian estimator consists of two steps. First, using the system model characterized by the transition probability $p(\underline{x}_k \,|\, \underline{x}_{k-1})$, we can *predict* the state at the time step k from the state at time step $k-1$ by

$$p(\underline{x}_k \,|\, \mathcal{Y}_{1:k-1}) = \int p(\underline{x}_k \,|\, \underline{x}_{k-1}) \cdot p(\underline{x}_{k-1} \,|\, \mathcal{Y}_{1:k-1}) \; \mathrm{d}\underline{x}_{k-1} \;.$$

Second, the state is *updated* with the received set of measurements $\mathcal{Y}_k$ using (8), by

$$p(\underline{x}_k \,|\, \mathcal{Y}_{1:k}) \propto p(\mathcal{Y}_k \,|\, \underline{x}_k) \cdot p(\underline{x}_k \,|\, \mathcal{Y}_{1:k-1}) \;. \tag{9}$$

V. Interacting Shape Models

In this section, we implement the proposed ideas as a recursive Bayesian estimator based on nonlinear Kalman filters. The advantage of this approach is its simplicity, ease of implementation and model prototyping, and high performance.

The method in this paper draws from related approaches dealing with random and Markov switch models to approximate (6), such as *Generalized Pseudo-Bayes* (GPB) [15] or *Interacting Multiple Models* (IMM) [16]. The idea is, first, to execute the prediction and update steps for each component, described in Sec. V-A, and then Sec. V-B will show how to merge the results.

This procedure gives the approach a measure of modularity, given that each component works independently and only the end results are fused. This allows a combination of techniques to be used depending on the needs of each component, ranging from analytical solutions and sample-based approximations [24], [25], to more general Gaussian filters not based on the Kalman formulas [26]. For simplicity, it will be assumed that only a single measurement is processed each time step, i.e., $\mathcal{Y}_k = \{ \underline{y}_k \}$. Multiple measurements can be processed sequentially, or be stacked together into a single, combined measurement.

A. Component Filtering Step

The idea behind nonlinear Kalman filtering is to use the framework provided by the Kalman formulas, and extend it to nonlinear measurement equations to achieve an approximate solution. Each component m maintains a state $\underline{\boldsymbol{x}}_k^m$, assumed to be Gaussian distributed in the form of

$$p(\underline{x}_k^m) = \mathcal{N}(\underline{\hat{x}}_k^m, \mathbf{C}_k^{x,m}) \;. \tag{10}$$

The prediction step is described by (3), in function of the last state $\underline{\boldsymbol{x}}_{k-1}^m$, yielding the predicted state

$$\underline{\boldsymbol{x}}_k^{p,m} = \underline{a}^m(\underline{\boldsymbol{x}}_{k-1}^m, \underline{\boldsymbol{w}}_k^m) \;. \tag{11}$$

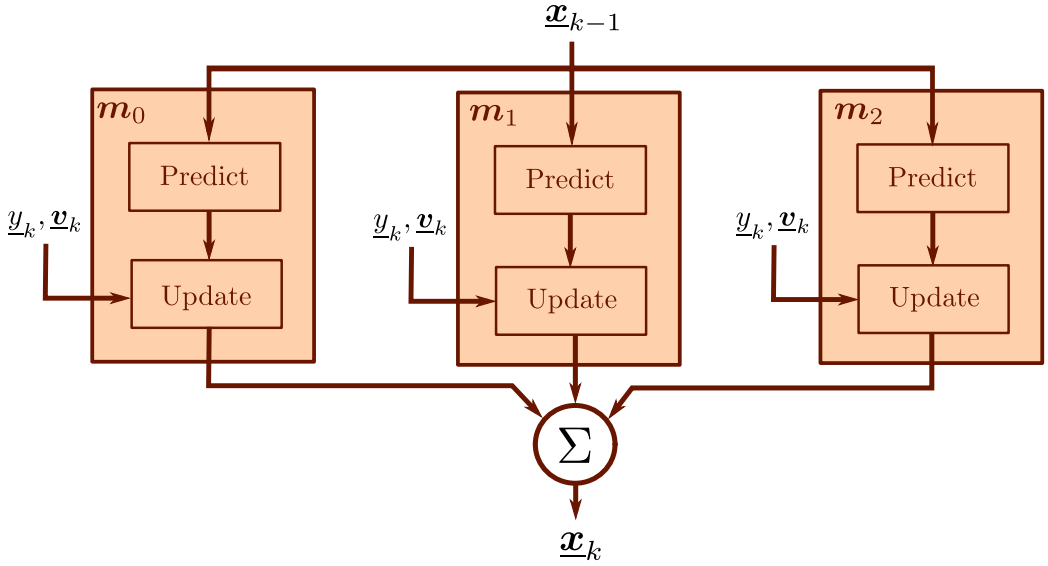


Figure 5: Filter for combined shape models using a random switch model.

with mean $\hat{\underline{x}}_k^{p,m}$ and covariance matrix $\mathbf{C}_k^{x,p,m}$, so that

$$\underline{\boldsymbol{x}}_k^{p,m} \sim p(\underline{x}_k \,|\, m, \mathcal{Y}_{1:k-1}) = \mathcal{N}(\hat{\underline{x}}_k^{p,m}, \mathbf{C}_k^{x,p,m}) \ . \tag{12}$$

The update step for $\underline{y}_k$ is implemented according to (2), leading to the random variable

$$\underline{\boldsymbol{y}}_k^{h,m} := h(\underline{\boldsymbol{x}}_k^{p,m}, \underline{y}_k, \underline{\boldsymbol{v}}_k, \boldsymbol{s}_k^m) \ . \tag{13}$$

From $\underline{\boldsymbol{y}}_k^{h,m}$, the mean $\hat{\underline{y}}_k^{h,m}$, covariance matrix $\mathbf{C}_k^{h,m}$, and cross-covariance matrix $\mathbf{C}_k^{xh,m}$ are obtained. Finally, the updated state is obtained by using the Kalman formulas, relating the pseudo-measurement $\underline{\boldsymbol{y}}_k^{h,m}$ to $\underline{0}$. Thus, we obtain the gain

$$\mathbf{K}_k^m := \mathbf{C}_k^{xh,m} \cdot (\mathbf{C}_k^{h,m})^{-1} \ ,$$

which leads to the updated state

$$\underline{\boldsymbol{x}}_k^m \sim p(\underline{x}_k \,|\, m, \mathcal{Y}_{1:k}) = \mathcal{N}(\hat{\underline{x}}_k^m, \mathbf{C}_k^{x,m}) \ , \tag{14}$$

with mean

$$\hat{\underline{x}}_k^m = \hat{\underline{x}}_{k|k-1}^{p,m} - \mathbf{K}_k^m \cdot \hat{\underline{y}}_k^h \ ,$$

and covariance matrix

$$\mathbf{C}_k^{x,m} = \mathbf{C}_k^{x,p,m} - \mathbf{K}_k^m \cdot \mathbf{C}_k^{h,m} \cdot (\mathbf{K}_k^m)^T \ .$$

Note that, depending on the model, there might be no need to do the prediction step in each individual filter as in (11). For simple system models, a *direct prediction step* for the combined model as in (3) is also possible.

B. Processing the Mixture

In order to merge the results from Sec. V-A, we still need a proper association model, as shown in (5). A way to obtain an association model for the component m, related to the measurement $\underline{y}_k$, is by using (7), leading to

$$\pi_k^m := p(\underline{y}_k \,|\, m) \cdot p(m) \ ,$$

where, using the terms derived from (13), we obtain

$$p(\underline{y}_k \,|\, m) \approx \mathcal{N}(\underline{0}; \hat{\underline{y}}_k^{h,m}, \mathbf{C}_k^{h,m}) \ ,$$

which, after normalization, produces the weighted association model

$$p(m \,|\, \underline{y}_k) = \frac{\pi_k^m}{\sum\limits_{m^* \in \mathcal{M}} \pi_k^{m^*}} \ . \tag{15}$$

For random switch models, merging is straightforward. From (9), plugging in (5), we obtain the combined prediction and update steps

$$p(\underline{x}_k \,|\, \mathcal{Y}_{1:k}) \propto \sum_{m \in \mathcal{M}} p(\underline{x}_k \,|\, m, \mathcal{Y}_{1:k}) \cdot p(m \,|\, \underline{y}_k) \ ,$$

using the terms from (14) and (15). In other words, each component filter is initialized with $\underline{\boldsymbol{x}}_{k-1}^m = \underline{\boldsymbol{x}}_{k-1}$, the state is predicted and updated in parallel, and then the resulting Gaussian distributions are mixed. This process is visualized in Fig. 5. Finally, as required by (10), a Gaussian form for $\underline{\boldsymbol{x}}_k$ is obtained from the mixture $p(\underline{x}_k \,|\, \mathcal{Y}_{1:k})$ by moment matching.

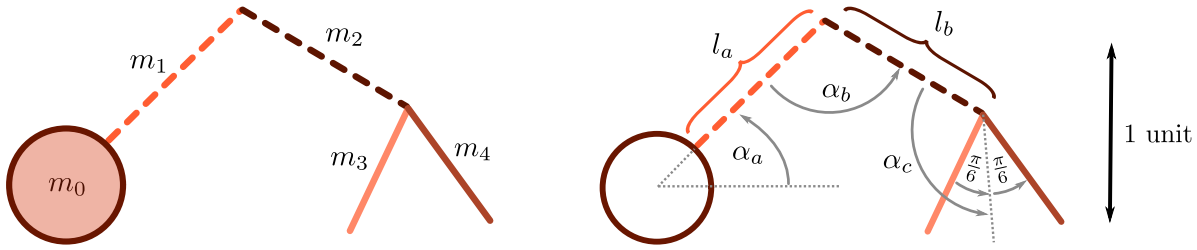
It can be seen that, for random switch models, the resulting filter is similar to a GPB-1 estimator. For Markov switch models, an extension is also straightforward using estimators based on IMM or GPB-2, derived using the association model from (15).

C. Data Validation

The usual mechanisms for data validation are *gating* and *clutter models*.

Gating makes a binary decision of whether to accept or reject the measurement $\underline{y}_k$ based on a user-defined parameter $\epsilon_k \in [0, 1]$. This can be easily implemented as part of the update step in each component. First, for each component m, (13) yields the mean $\hat{\underline{y}}_k^{h,m}$ and the covariance matrix $\mathbf{C}_k^{h,m}$. The squared Mahalanobis distance to the component m is then defined as

$$\eta_k^m = (\hat{\underline{y}}_k^{h,m})^T \cdot (\mathbf{C}_k^{h,m})^{-1} \cdot \hat{\underline{y}}_k^{h,m} \ .$$



(a) List of components. (b) State parameters.

Figure 6: Combined shape object to be tracked.

η_k^m assumed to be distributed according to $\chi^2(H)$, where H denotes the dimension of the vector $\hat{\underline{y}}_k^{h,m}$. Then, we say that the component m *validates* the measurement $\underline{y}$ if

$$P_{\chi^2}(\eta_k^m) < \epsilon_k \ ,$$

where P_{χ^2} represents the CDF of $\chi^2(H)$. The validations of each component can be combined for determining whether the combined shape model accepts the measurement. For example, the combined shape model may accept a measurement if at least one component validates it.

A clutter model can also be easily implemented by creating a dummy clutter component m^C, with corresponding probability $p(m^C \,|\, \underline{y})$ that the measurement is invalid. The prediction (12) and update (14) steps for this component would be the identity.

VI. Evaluation

The evaluation consisted of tracking the object shown in Fig. 6 using synthetic measurements.

A. Tracked Target

The combined shape model is divided into five components, as shown in Fig. 6a, and parameterized as described in Fig. 6b. For the component m_0, the shape is a circular disc, and for every other component the shape is a line segment. The ground truth lengths are $l_a^G = l_b^G = 1$ unit, while the ground truth angles changed in time. The remaining values not shown in Fig. 6b are fixed, and known to the filter. The radius of m_0 is 0.3 units, and the lengths of the segments m_3 and m_4 are 0.75 units. The disc from m_0 is centered on the origin.

The ground truth shape is moving as if *flexing*, or contracting, with the angles changing sinusoidally, i.e., in the form of $\sin(10^{-3} \cdot k)$. The flex start is pictured in Fig. 7b, at around time step 1600, and the end is in Fig. 7c, at around time step 4700. The motion is periodic and repeats itself after about 6300 steps. At the beginning of the evaluation, the target is in a configuration similar to that shown in Fig. 6.

B. Estimator State

The parameters to be estimated are described in Fig. 6b. More specifically, the state $\underline{x}_k$ can be divided in three blocks, in the form of

$$\underline{x}_k = \left[\ \underline{x}_{a,k}, \underline{x}_{b,k}, \underline{x}_{c,k} \ \right] \ ,$$

with the first block, related to m_1, having the form

$$\underline{x}_{a,k} = \left[\ l_{a,k}, \alpha_{a,k}, \omega_{a,k} \ \right] \ ,$$

the second block related to m_2,

$$\underline{x}_{b,k} = \left[\ l_{b,k}, \alpha_{b,k}, \omega_{b,k} \ \right] \ ,$$

and, finally, the third block related to m_3 and m_4,

$$\underline{x}_{c,k} = \left[\ \alpha_{c,k}, \omega_{c,k} \ \right] \ .$$

The component parameters represent the segment length, angle, and angular velocity. This leads to a state $\underline{x}_k$ with eight parameters.

C. Component Measurement Functions

For the sake of completeness, this section describes the measurement functions used by the components.

For m_0, a circular RHM is used. This leads to the measurement function for m_0, derived from [3], as

$$h^{m_0}(\underline{x}_k, \underline{y}_k, \underline{v}_k, s_c) = |\, \underline{y}_k - \underline{b} \,|^2 - (r \cdot s_c)^2 - \bar{v}_k \ ,$$

where $\underline{b}$ is the origin, r has a constant value of 0.3, and $\bar{\boldsymbol{v}}_k$ is a bias correction term derived from $\underline{\boldsymbol{v}}_k$. In particular, for isotropic measurement noise in the form $\mathbf{C}^v = \sigma^2 \cdot \mathbf{I}$, [4] shows that it holds that $\bar{\boldsymbol{v}}_k \sim \mathcal{N}(2\sigma^2, 4\sigma^4)$. The shape noise parameter s_c is drawn from $\boldsymbol{s}_c$, where $\boldsymbol{s}_c^2 \sim \mathcal{U}(0,1)$.

For the other components m_i, for $1 \leq i \leq 4$, a spatial distribution is used to describe the segments. Each line segment i can be described in function of a length $l_{i,k}$, a direction $\beta_{i,k}$, and a starting point $\underline{p}_{i,k}$. It can be interpreted as the transformation of a base segment spanning from $[0,0]^T$ to $[l_{i,k}, 0]^T$, which is then rotated by $\beta_{i,k}$, and then translated by $\underline{p}_{i,k}$. By applying the inverse of the last two transformations on $\underline{y}_k$, we can derive a measurement function using the base segment as

$$h^{m_i}(\underline{x}_k, \underline{y}_k, \underline{v}_k, s_\ell) = \begin{bmatrix} l_{i,k} \cdot s_\ell \\ 0 \end{bmatrix} - \mathbf{R}(\beta_{i,k})^{-1}(\underline{y}_k - \underline{v}_k - \underline{p}_{i,k}) \ ,$$

where $\mathbf{R}(\beta_{i,k})$ is a rotation matrix in the form of

$$\mathbf{R}(\beta_{i,k}) := \begin{bmatrix} \cos(\beta_{i,k}) & -\sin(\beta_{i,k}) \\ \sin(\beta_{i,k}) & \cos(\beta_{i,k}) \end{bmatrix} \ ,$$

and s_ℓ is drawn from $\boldsymbol{s}_\ell \sim \mathcal{U}(0,1)$, which serves to index each measurement source of the base segment. The terms $\beta_{i,k}$, $\underline{p}_{i,k}$ and $l_{i,k}$ are derived from the state parameters by taking into account Fig. 6b, and the data from Sec. VI-A.

D. Experiments

The evaluation is divided in two scenarios, *Scenario A* with high measurement noise and *Scenario B* with low measurement noise. The component distribution $p(m)$ was assumed to be uniform. In addition, the distribution of measurement sources in each component shape was also assumed as uniform. Only the weighted association model was considered, given that, for the noise levels used, a direct association invariably caused the estimator to diverge. For the component filters, an S^2KF [25] was used with a total of 32 samples.

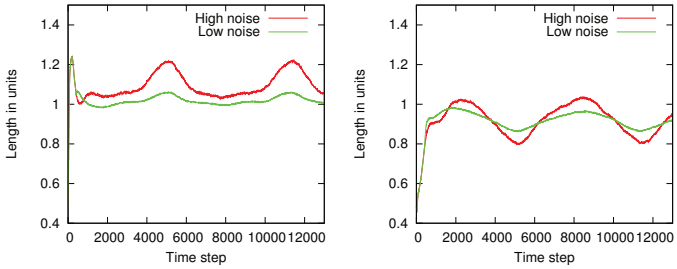
(a) Initialization. (b) Flex start. (c) Flex end.

Figure 7: Used configurations.

The starting state is $\underline{\boldsymbol{x}}_0 \sim \mathcal{N}(\underline{\hat{x}}_0, \mathbf{C}_0^x)$. For both scenarios, the state mean is initialized as shown in Fig. 7a, with the lengths as $\hat{l}_{a,0} = \hat{l}_{b,0} = 0.5$ units. The starting angles are set to $\frac{\pi}{2}$, and the starting velocities are set to 0. The starting state covariance matrix $\mathbf{C}_0^x$ is $10^{-3} \cdot \mathbf{I}$ for Scenario A, and $10^{-4} \cdot \mathbf{I}$ for Scenario B.

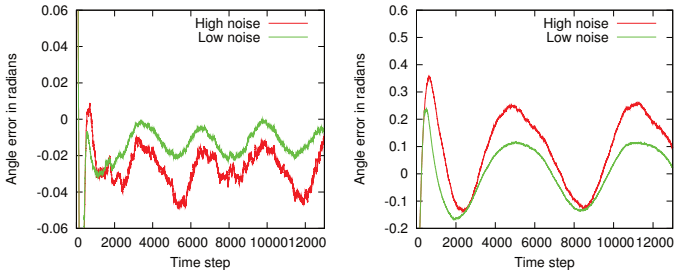
At each time step k, a component is drawn according to $p(m)$, and from the corresponding shape, a single measurement source is drawn. This measurement source is then corrupted as shown in (1), with $\mathbf{C}^v$ having a value of $10^{-2} \cdot \mathbf{I}$ for Scenario A, and $10^{-3} \cdot \mathbf{I}$ for Scenario B. The system model consists on a constant velocity model, leading to a direct linear prediction step. As described before, this system model does not correctly consider the sinusoidal velocities, which serves to further test the capabilities of the estimator. Finally, an additive process noise $\underline{\boldsymbol{w}}_k \sim \mathcal{N}(\underline{0}, \mathbf{C}^w)$ is assumed, with $\mathbf{C}^w$ having a value of $10^{-5} \cdot \mathbf{I}$ for Scenario A, and $10^{-6} \cdot \mathbf{I}$ for Scenario B.

Fig. 10 shows snapshots of the evaluation. Note that the multiple example measurements are for visualization, for the estimator only processes a single measurement per time step. Fig. 8 and Fig. 9 show the results of the evaluation, averaged for 100 runs. The generating components, measurement sources, and noise parameters are drawn independently for each run.



(a) Length for l_a. (b) Length for l_b.

Figure 8: Estimated segment lengths. Ground truth length is 1.



(a) Angle error for α_b. (b) Angle error for α_c.

Figure 9: Error of estimated angles.

The effects of the sinusoidal movement are evident. It can be seen that the errors increase at right after time steps 1600, 4700, 7800 and 11000, which is when the flexing motion changes direction, requiring the constant velocity model to readjust. In particular, the error around time steps 5000 and 11000 shows that the estimators are still recovering from the high association uncertainty described in Fig. 10a and Fig. 10d. Still, it is clear that the tracker can follow the target object, even if each component estimator only considers part of the whole shape.

VII. Conclusion

As sensor capabilities improve, estimators can incorporate more information about complex shapes to improve estimation results. In this paper, we presented an approach to model complex shapes as the combination of simple shapes. In order to achieve this, the approach drew from works in multiple object tracking and dynamic multiple model estimation. The key idea was to divide the target into components with individual shape models, potentially with different filtering techniques as necessary. Thus, the estimation procedure consisted of incorporating measurements in each individual component, and then combining the results, allowing for modularity and encapsulation.

The approach was implemented using a combination of nonlinear Kalman filters. Then, the estimator was evaluated using an armature target object, which was subdivided into simpler components. The results showed that the proposed approach could handle high association uncertainty, and could track the target even with high noise.

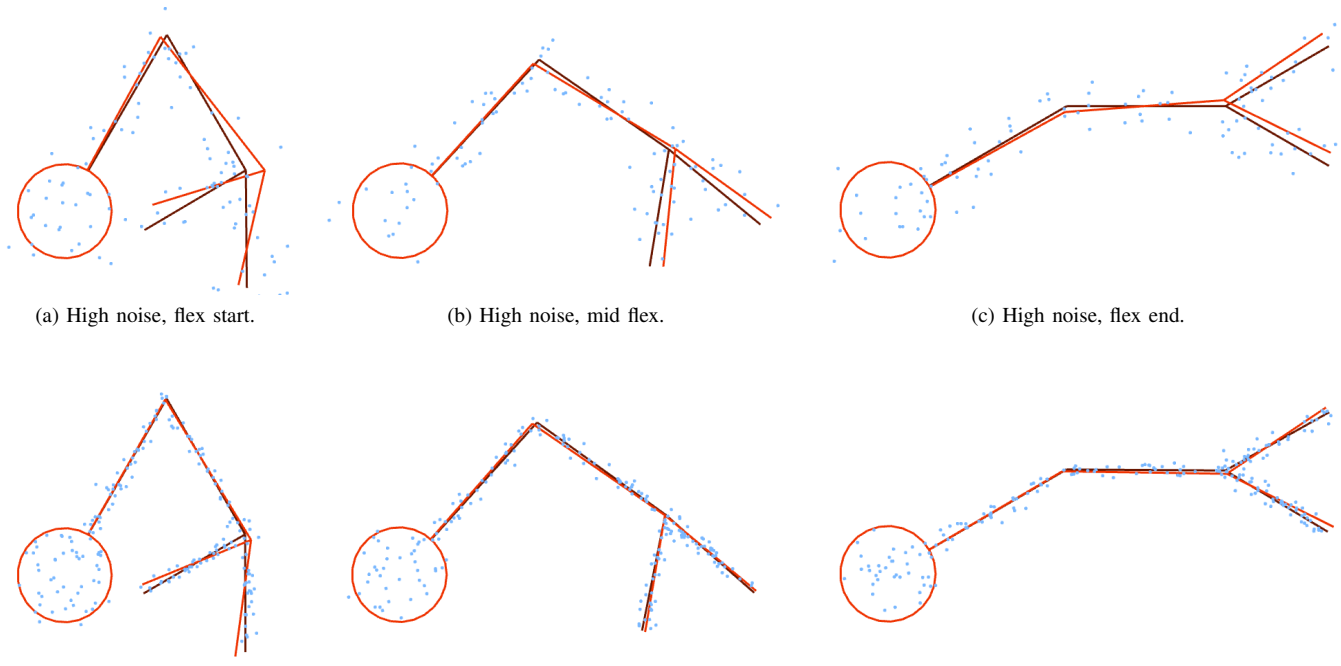
References

[1] Z. Zhang, "Parameter estimation techniques: a tutorial with application to conic fitting," *Image and Vision Computing*, vol. 15, no. 1, pp. 59 – 76, 1997.

[2] K.-i. Kanatani, "Statistical bias of conic fitting and renormalization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 3, pp. 320–326, 1994.

[3] M. Baum and U. D. Hanebeck, "Fitting Conics to Noisy Data Using Stochastic Linearization," in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2011)*, San Francisco, California, USA, Sep. 2011.

[4] M. Baum, V. Klumpp, and U. D. Hanebeck, "A Novel Bayesian Method for Fitting a Circle to Noisy Points," in *Proceedings of the 13th International Conference on Information Fusion (Fusion 2010)*, Edinburgh, United Kingdom, 2010.



(a) High noise, flex start. (b) High noise, mid flex. (c) High noise, flex end.

(d) Low noise, flex start. (e) Low noise, mid flex. (f) Low noise, flex end.

Figure 10: Snapshots of evaluation runs. Example measurements in blue, ground truth in dark red, estimate in orange.

[5] M. Feldmann, D. Franken, 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.

[6] K. Gilholm and D. Salmond, "Spatial Distribution Model for Tracking Extended Objects," *IEE Proceedings on Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 364–371, 2005.

[7] 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.

[8] A. Blake and M. Isard, *Active Contours: The Application of Techniques from Graphics,Vision,Control Theory and Statistics to Visual Tracking of Shapes in Motion*, 1st ed. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1998.

[9] 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.

[10] A. Zea, F. Faion, M. Baum, and U. D. Hanebeck, "Level-Set Random Hyper Surface Models for Tracking Complex Extended Objects," in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, 2013.

[11] M. Baum, B. Noack, and U. D. Hanebeck, "Random Hypersurface Mixture Models for Tracking Multiple Extended Objects," in *Proceedings of the 50th IEEE Conference on Decision and Control (CDC 2011)*, Orlando, Florida, USA, Dec. 2011.

[12] K. Granstrom, C. Lundquist, and U. Orguner, "A gaussian mixture phd filter for extended target tracking," in *Information Fusion (FUSION), 2010 13th Conference on*, July 2010, pp. 1–8.

[13] R. Mahler, "Phd filters for nonstandard targets, i: Extended targets," in *Information Fusion, 2009. FUSION'09. 12th International Conference on*. IEEE, 2009, pp. 915–921.

[14] S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 19, no. 1, pp. 5–18, Jan 2004.

[15] G. Ackerson and K. Fu, "On state estimation in switching environments," *Automatic Control, IEEE Transactions on*, vol. 15, no. 1, pp. 10–17, 1970.

[16] H. A. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with markovian switching coefficients," *Automatic Control, IEEE Transactions on*, vol. 33, no. 8, pp. 780–783, 1988.

[17] 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.

[18] P. Lancaster and K. Salkauskas, "Curve and surface fitting. an introduction," *London: Academic Press, 1986*, vol. 1, 1986.

[19] M. Baum and U. D. Hanebeck, "Random Hypersurface Models for Extended Object Tracking," in *2009 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*. Ajman, United Arab Emirates: Ieee, Dec. 2009, pp. 178–183.

[20] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Robotics-DL tentative*. International Society for Optics and Photonics, 1992, pp. 586–606.

[21] S. Knoop, S. Vacek, and R. Dillmann, "Sensor fusion for 3d human body tracking with an articulated 3d body model," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, May 2006, pp. 1686–1691.

[22] F. Scholz, "Maximum likelihood estimation," *Encyclopedia of Statistical Sciences*, 1985.

[23] H. Sorenson, *Parameter estimation: principles and problems*, ser. Control and systems theory. M. Dekker, 1980.

[24] E. A. Wan and R. Van Der Merwe, "The unscented kalman filter for nonlinear estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*. IEEE, 2000, pp. 153–158.

[25] J. Steinbring and U. D. Hanebeck, "S2KF: The Smart Sampling Kalman Filter," in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, Jul. 2013.

[26] U. D. Hanebeck and J. Steinbring, "Progressive Gaussian Filtering Based on Dirac Mixture Approximations," in *Proceedings of the 15th International Conference on Information Fusion (Fusion 2012)*, Singapore, Jul. 2012.