Exploiting Clutter: Negative Information for Enhanced Extended Object Tracking

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

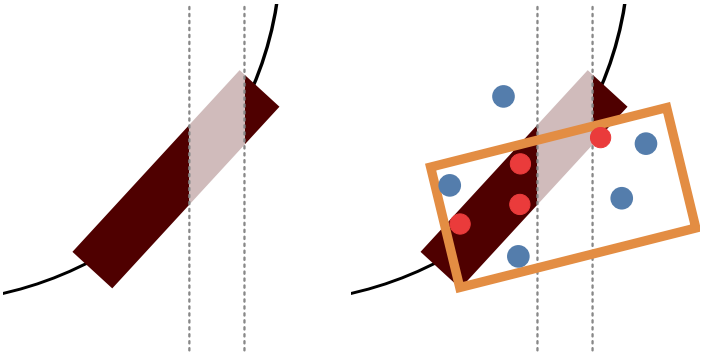
***Abstract*—When tracking an extended object, traditional approaches exploit information only from measurements that are assumed to stem from the target, and discard observations assumed to have been generated elsewhere. However, the fact that these observations were received contains valuable information about where the target is not. This information, which is usually treated as clutter with little value, can also be exploited in order to improve estimation results. This becomes particularly important in situations with low measurement quality or occlusions, where positive observations from the target may be scarce. In these cases, negative observations, which show where the target cannot be, become highly valuable. In this paper, we introduce *Silhouette Models*, which are able to incorporate information from both types of observations. The benefits of this approach, which include more robust results and resistance to occlusion, are confirmed in the evaluation.**

Keywords*—*Extended object tracking, negative observations, negative information, shape models, shape estimation, silhouettes.

I. Introduction

The traditional approach to track an object is by assuming that measurements are generated by a single point source. However, the increasing resolution of modern sensors requires the consideration of measurements that originate from different points on the target's boundary. In these situations, more robust results can be obtained by considering the shape of the target in addition to its pose. These approaches are part of the field known as extended object tracking. A critical part of these estimators is the probabilistic treatment of the shape parameters. On the one hand, they can be categorized by the representation used, such as lines [1], ellipses [2], [3], polynomials [4] or extended Gaussian images [5]. On the other hand, they can be described through their source models. Some assume a probability distribution over sources [1], [3], [6], others consider only a single source in the form of a projection [7]–[9], and some model sources as the result of drawing from a random variable [2], [10].

Most of these works have in common that they only take into account positive observations, i.e., measurements that are assumed to be generated from the target. However, sensors typically not only obtain information about where the target *is*, but also about where the target *is not*. For example, radars using MTI or RGBD cameras inevitably produce negative observations that are known, with high certainty, not to belong to the target object. This valuable information, which can help produce a more reliable and robust estimate, is usually discarded as *clutter* without much consideration. Fig. 1 shows a motivating example.



(a) Motivating scenario. (b) Positive and negative observations.

Figure 1: In the left figure, a target (brown rectangle) is moving along a path (black), but a region is being unexpectedly occluded (white stripe) and all measurements from there are unavailable. To the right, the tracker fails to recognize the rotation as the incorrect estimate (orange) covers all positive observations (red). Exploiting negative observations (blue) would lead to a more robust result.

To address this issue, we will derive an estimator capable of incorporating both positive and negative observations. In order to achieve this, we will model how both kinds of measurements are generated and use these concepts to derive a likelihood function. The resulting approach is called *Silhouette Models*, as we will determine probabilistically which regions in space are occupied by the target shape and which ones are not. The derived likelihood can then be used in a Bayesian estimator.

Related work can be divided into three categories. The first category is extended object tracking, a topic which includes the works previously mentioned [1]–[10]. The second category is 'negative' information, as treated in [11]–[13]. Note, however, that in this paper we do not deal with *absent* measurements, but instead with measurements that do not belong to the target. Finally, some of the ideas presented in this paper are related to the category of constraint information from projections or silhouettes, also handled in [14]–[16].

The structure of this paper is as follows. First, the problem formulation is specified in Sec. II. Then, some traditional models to describe extended objects with positive observations are described in Sec. III. This is followed by the introduction of our *Silhouette Models* in Sec. IV. An example implementation is described in Sec. V and evaluated in Sec. VI. We conclude the paper with Sec. VII.

II. Problem Formulation

The task explored in this paper is estimating the parameters $\underline{x}$ of an extended target, such as its shape and pose, based on incoming point measurements. The shape of the extended target is denoted as the set of points $\mathcal{Z}^x$, and generates measurements both from its boundary and its interior. The incoming measurements are collected in $\mathcal{Y} = \{\underline{y}_0, \cdots, \underline{y}_l\}$, where each measurement $\underline{y} \in \mathcal{Y}$ is described by its components

$$\underline{y} = \left[y^t, \underline{y}^p\right] \ .$$

The component $\underline{y}^p$ represents a *position* in Cartesian coordinates drawn from $\mathbb{R}^d$, where usually $d \in \{1, 2, 3\}$. The value y^t represents the measurement *type*, which belongs to

$$\odot := \left\{\odot^+, \odot^-\right\} \ .$$

This value can be interpreted in the following way. If $y^t = \odot^+$, we say that $\underline{y}$ is a positive observation, i.e., we assume it was generated by the target. However, if $y^t = \odot^-$, we say that $\underline{y}$ is a negative observation, or in other words, we assume it was not generated by the target. Each measurement $\underline{y}$ is seen as having been generated using the following two steps (Fig. 2). In the first step, a source point $\underline{z}^p$ is selected from $\mathbb{R}^d$. The corresponding observation type z^t is determined as $\odot^+$ if $\underline{z}^p$ belongs to $\mathcal{Z}^x$, or $\odot^-$ if it does not (Fig. 2a). This yields the source $\underline{z} = \left[z^t, \underline{z}^p\right]$. We model this process using the probability density $p(\underline{z}|\underline{x})$. In the second step, the source $\underline{z}$ is corrupted by noise according to the sensor model, the result of which is the measurement $\underline{y}$. This noise has two aspects. On the one hand, the position $\underline{y}^p$ is assumed to be corrupted by additive zero-mean Gaussian noise with known covariance matrix $\mathbf{C}^v$ (Fig. 2b), resulting in

$$p(\underline{y}^p|\underline{z}^p) = \mathcal{N}\left(\underline{y}^p - \underline{z}^p; \underline{0}, \mathbf{C}^v\right) \ , \tag{1}$$

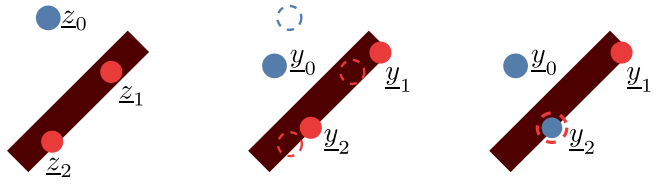
where $\mathbf{C}^v$ may be different for each measurement. On the other hand, sensor noise may also cause a misinterpretation of the observation type, causing it to flip from $\odot^+$ to $\odot^-$, or viceversa, illustrated in Fig. 2c. Note that we use the expression "sensor noise" as an umbrella term, as these errors can also originate from segmentation algorithms, classification approaches, or other uncertainties in the underlying sensor model.

Probabilistically, the generation of the measurement $\underline{y}$ can be described using the likelihood $p(\underline{y}|\underline{x})$, i.e., $p(y^t, \underline{y}^p|\underline{x})$. We assume the noise terms are independent from each other. Thus, for the set of received measurements $\mathcal{Y}$, the corresponding likelihood $p(\mathcal{Y}|\underline{x})$ can be written as

$$p(\mathcal{Y}|\underline{x}) = \prod_{\underline{y} \in \mathcal{Y}} p(y^t, \underline{y}^p|\underline{x}) \ , \tag{2}$$

so that each individual $p(\underline{y}|\underline{x})$ can be treated separately. For simplicity, we will focus on the case where $d = 2$, i.e., source and measurement positions are in $\mathbb{R}^2$. Note, however, that the presented concepts can be easily extended into other dimensions.

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 $\underline{y}_{i,k}$. However, as most of the discussions are related to the shape model, we will drop the subindex k for legibility unless needed.



(a) Selecting sources. (b) Noise on position. (c) Noise on type.

Figure 2: Generative model. First, sources are selected from $\mathbb{R}^2$, and their types are set as positive (red) if inside the target (brown), or negative (blue) otherwise. Then, sensor noise is applied that disturbs the positions and observation types.

III. Traditional Models

As a first step, we would like to review traditional extended object models that only use positive measurements. In other words, these models assume that the measurements and their sources always originate from the target $\mathcal{Z}^x$. Thus, when considering a measurement $\underline{y}$ or a source $\underline{z}$, we do not take the observation type into account, and instead, we only consider their positions $\underline{y}^p$ and $\underline{z}^p$. In the following, we will discuss *Spatial Distribution Models* (SDMs) and *Greedy Association Models* (GAMs), including a kernel extension for the latter. In particular, we will elaborate on a critical issue that affects GAMs. These models will serve as a basis for Sec. IV.

A. Spatial Distribution Models

A Spatial Distribution Model [1] assumes that $p(\underline{z}^p|\underline{x})$ is known a priori. This allows us to obtain a likelihood by combining (1) with $p(\underline{z}^p|\underline{x})$ and marginalizing $\underline{z}^p$, leading to

$$p(\underline{y}|\underline{x}) = p(\underline{y}^p|\underline{x}) = \int_{\mathbb{R}^2} p(\underline{y}^p|\underline{z}^p) \cdot p(\underline{z}^p|\underline{x}) \ \mathrm{d}\underline{z}^p \ . \tag{3}$$

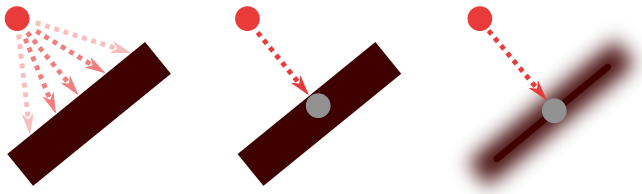
For illustration, an example of this model follows (Fig. 3a). For a given shape $\mathcal{Z}^x$, a common assumption is that all sources are equally likely to be selected [1], [3]. This leads to

$$\begin{aligned} p(\underline{z}^p|\underline{x}) &= \frac{1}{|\mathcal{Z}^x|} \cdot \begin{cases} 1, & \text{if } \underline{z}^p \in \mathcal{Z}^x \\ 0, & \text{otherwise,} \end{cases} \\ &= \frac{1}{|\mathcal{Z}^x|} \cdot \mathbf{1}_{\mathcal{Z}^x}(\underline{z}^p) \ , \end{aligned} \tag{4}$$

where $|\mathcal{Z}^x|$ represents the area of the shape $\mathcal{Z}^x$, and $\mathbf{1}_{\mathcal{Z}^x}(\cdot)$ is the indicator function of the shape. Next, the source is corrupted using Gaussian noise as described in (1). Finally, plugging (4) and (1) in (3) leads to

$$p(\underline{y}^p|\underline{x}) = \frac{1}{|\mathcal{Z}^x|} \cdot \int_{\mathcal{Z}^x} \mathcal{N}\left(\underline{y}^p - \underline{z}^p; \underline{0}, \mathbf{C}^v\right) \ \mathrm{d}\underline{z}^p \ . \tag{5}$$

These likelihoods generally have two issues. On the one hand, integrating over the set $\mathcal{Z}^x$ may be intractable for general shapes such as non-convex filled shapes. On the other hand, it might be difficult to know $p(\underline{z}^p|\underline{x})$ a priori. This is a problem in real-life scenarios, where due to unexpected occlusions or related artifacts it becomes impossible to tell which sources are visible, and a uniform distribution may not be assumed.



(a) SDM. (b) GAM. (c) Kernel GAM.

Figure 3: Comparison of shape models.

B. Greedy Association Models

A different approach, used when $p(\underline{z}^p|\underline{x})$ is unknown, is to model the generating source explicitly. For this, we use a *projection function* $\pi_x(\underline{y}^p)$, which calculates the point in $\mathcal{Z}^x$ that minimizes some distance measurement or optimality criterion (e.g., the Euclidian distance) related to $\underline{y}^p$. Probabilistically, we say that the only possible source is the projection $\pi_x(\underline{y}^p)$, i.e.,

$$p(\underline{z}^p|\underline{x}) \approx \delta\left(\underline{z}^p - \pi_x(\underline{y}^p)\right) \ , \tag{6}$$

where $\delta(\cdot)$ represents the Dirac-delta distribution. Then, we plug (1) and (6) into (3), yielding

$$\begin{aligned} p(\underline{y}^p|\underline{x}) &= \int_{\mathbb{R}^2} \mathcal{N}\left(\underline{y}^p - \underline{z}^p; \underline{0}, \mathbf{C}^v\right) \cdot \delta\left(\underline{z}^p - \pi_x(\underline{y}^p)\right) \ \mathrm{d}\underline{z}^p \\ &= \mathcal{N}\left(\underline{y}^p - \pi_x(\underline{y}^p); \underline{0}, \mathbf{C}^v\right) \ . \end{aligned} \tag{7}$$

We denote approaches like this, which can be seen as a probabilistic interpretation of distance minization techniques [1], as Greedy Association Models (Fig. 3b).

C. Kernel GAMs

In situations with low measurement quality or unexpected occlusions, using complex shape models may become detrimental as they are prone to artifacts such as overfitting. In these cases, simpler shapes such as line segments or ellipses are preferred as they allow for more robust results, even if measurements are geometrically in a higher dimension, e.g. in $\mathbb{R}^3$. However, this simplified model cannot fully represent the sources present in the target shape. In order to compensate for this, the extent of the shape may be modeled by means of random variables. This approach is widely used in literature, for example as multiplicative noise [10], random matrices [6], or as a random transformation of a boundary [2].

A simple implementation can be obtained by applying a Gaussian kernel onto the shape, in the form of a convolution. Fig. 3c shows an example where a Gaussian kernel was applied on the line segment $\mathcal{Z}^x$. Probabilistically, the kernel can be interpreted in a similar way as additive Gaussian noise. Thus, generating a source $\underline{z}^p$ can be seen as, first, selecting a source from $\mathcal{Z}^x$, and then distorting it with additive zero-mean Gaussian noise of covariance matrix $\mathbf{C}_x^K$. This source is subsequently distorted again with sensor noise to obtain the measurement $\underline{y}^p$. An advantage of this approach is that the kernel parameters can be easily estimated from incoming measurements as part of the state $\underline{x}$, and interpreted as part of the target extent. To show the simplicity of this approach, we will now extend GAMs to use Gaussian kernels. The source selection can be modeled by modifying (6) as

$$p(\underline{z}^p|\underline{x}) \approx \mathcal{N}\left(\underline{z}^p - \pi_x(\underline{y}^p); \underline{0}, \mathbf{C}_x^K\right) \ , \tag{8}$$

or in other words, the kernel is applied directly onto the projection. When plugged into (3), this yields the likelihood

$$\begin{aligned} p(\underline{y}^p|\underline{x}) &= \int_{\mathbb{R}^2} \mathcal{N}\left(\underline{y}^p - \underline{z}^p; \underline{0}, \mathbf{C}^v\right) \cdot \mathcal{N}\left(\underline{z}^p - \pi_x(\underline{y}^p); \underline{0}, \mathbf{C}_x^K\right) \ \mathrm{d}\underline{z}^p \\ &= \mathcal{N}\left(\underline{y}^p - \pi_x(\underline{y}^p); \underline{0}, \mathbf{C}_x^K + \mathbf{C}^v\right) \ , \end{aligned} \tag{9}$$

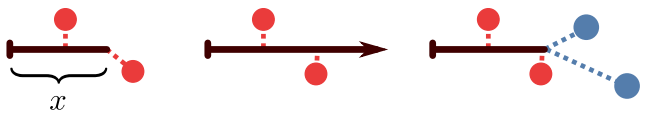
i.e., the likelihood interprets the kernel as nothing more than additional noise. We denote these models as Kernel GAMs.

D. Discussion of GAMs and SDMs

We observe that the derivations of GAMs and SDMs are very similar. However, they stand in contrast to each other in the following ways. On the one hand, GAMs and Kernel GAMs only associate the measurement with the "closest" source $\pi_x(\underline{y}^p)$. On the other hand, SDMs associate the measurement with all possible sources, which requires explicitly modeling $p(\underline{z}|\underline{x})$ beforehand. Thus, the likelihood for GAMs is much simpler and easier to implement. This comes with the caveat, however, that the projection is assumed to be a good approximation of the true source. Under high noise levels, $\pi_x(\underline{y}^p)$ cannot approximate the true source appropriately, which then leads to high estimation *bias* [9]. Another difference between both models is that, unlike SDMs, GAMs have a problem with overestimating the extent of the shape. This issue is described in detail in III-E.

E. GAMs and Extent Penalties

In general, we are only interested in the smallest shape that covers the received measurements. However, a problem present in GAMs and Kernel GAMs, as discussed in [1], appears when different possible states produce the same measurements. This issue is illustrated in Fig. 4, where the scalar state x of a line (brown) determines its length. In Fig. 4a, the estimator can reduce the distance to the rightmost positive observation by increasing the length. However, if the length is estimated too large, there is nothing to correct it to its shorter form, as this incorrect state also has a minimal distance to all measurements (Fig. 4b). This stands in contrast to SDMs as seen in (4), where dividing by the area penalizes larger extents automatically.

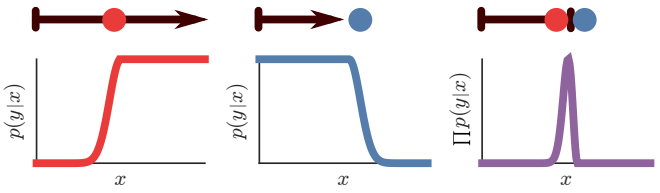


(a) Positive observations. (b) Extent problem. (c) Negative observations.

Figure 4: Positive observations that tell us where the shape *is*, but this leaves large extents unpenalized. In contrast, negative observations tell us where the shape is *not*.

We propose to address this by also incorporating negative observations (Fig. 4c), which tell the estimator to adjust the state so as *not* to cover them. The length issue can be visualized by plotting the likelihood (7) of a measurement $\underline{y}$

in terms of the line length x. The idea is as follows. If $\underline{y}$ is a positive observation (red), the estimator rewards line lengths that contain it, but larger lengths are still not discouraged (Fig. 5a). However, if the measurement is a negative observation (blue), the estimator will assume the opposite, i.e., states that cover the measurement are penalized and shorter lines are preferred (Fig. 5b). Thus, once all likelihoods are multiplied in (2), we can obtain a better idea of what the true extent is (Fig. 5c). This idea is elaborated in detail in Sec. IV.



(a) Likelihood (positive). (b) Likelihood (negative). (c) Likelihood product.

Figure 5: Idea sketch. We want positive observations to reward states that produce shapes near them, and negative observations to penalize them. The likelihood product reveals the true length.

IV. Silhouette Models

In this section, we aim to develop a probabilistic model for taking into account both positive and negative observations, which we denote as *Silhouette Models*. This will help us to not only exploit more information for SDMs, but also to work with GAMs without needing to worry about length issues. Dealing with the likelihood $p(y^t, \underline{y}^p|\underline{x})$ requires considering its two aspects, i.e., $p(\odot^+, \underline{y}^p|\underline{x})$ for positive observations, and $p(\odot^-, \underline{y}^p|\underline{x})$ for negative observations. As mentioned in [11], [13], this requires in turn some refinements in the shape and sensor models in order to accomodate for both likelihoods. Thus, in this section we will discuss both steps of the generative model, i.e., the source selection and the sensor noise. Finally, we will extend SDMs and Kernel GAMs to allow them to incorporate negative observations.

A. Selecting the Source

The source selection is determined by the probability density $p(\underline{z}|\underline{x})$, which can be rewritten as

$$\begin{aligned} p(\underline{z}|\underline{x}) &= p(z^t, \underline{z}^p|\underline{x}) \\ &= p(\underline{z}^p|\underline{x}) \cdot p(z^t|\underline{z}^p, \underline{x}) \ , \end{aligned} \tag{10}$$

where $p(\underline{z}^p|\underline{x})$ represents the probability of the position $\underline{z}^p$ being selected, and $p(z^t|\underline{z}^p, \underline{x})$ denotes the *discrete* probability that the observation in the given position $\underline{z}^p$ has the type z^t. For simplicity, we make the assumption that all positions are equally likely to be observed. We believe that this is a reasonable assumption, as devices such as time-of-flight sensors or depth cameras sample all positions available (e.g., pixels or grid cells). Thus, information about the target is not encoded in the measurement positions, but in the measurement types instead. We denote the set of all possible source positions as the *observed space* $\mathcal{S} \subset \mathbb{R}^2$. It follows that

$$p(\underline{z}^p|\underline{x}) = \underbrace{\frac{1}{|\mathcal{S}|}}_{:=c_S} \cdot \begin{cases} 1, & \text{if } \underline{z}^p \in \mathcal{S} \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where the term $|\mathcal{S}|$ is the area of the observed region. As seen later, in most cases it is not necessary to model $\mathcal{S}$ explicitly, but we assume it to be finite. Then, by plugging (11) in (10), it follows for $\underline{z}^p \in \mathcal{S}$ that

$$p(\underline{z}|\underline{x}) = c_S \cdot p(z^t|\underline{z}^p, \underline{x}) \ . \tag{12}$$

We note that, in order to describe a shape, we only need to concern us with modeling the distribution $p(\odot^+|\underline{z}^p, \underline{x})$. We denote this term as the *silhouette function*, an example of which can be seen in Fig. 6a. As $p(z^t|\underline{z}^p, \underline{x})$ is a discrete distribution, the remaining probability for $\odot^-$ can be obtained from

$$p(\odot^-|\underline{z}^p, \underline{x}) = 1 - p(\odot^+|\underline{z}^p, \underline{x}) \ .$$

B. Sensor Noise

The obtained source is then distorted with sensor noise, modeled by $p(\underline{y}|\underline{z})$. This term can be subdivided into the noise term that affects the observation type, described by $p(y^t|z^t)$, and the term which corrupts the position, specified by $p(\underline{y}^p|\underline{z}^p)$. We model both terms to be independent from each other, so that it holds

$$p(\underline{y}|\underline{z}) = p(y^t|z^t) \cdot p(\underline{y}^p|\underline{z}^p) \ . \tag{13}$$

In this paper, we will assume that the sensor does not directly corrupt the observation type, i.e., $y^t = z^t$. However, we will still obtain positive observations outside of the shape, and negative observations inside of it, as a consequence of the displacement by the position noise. Thus, we simplify (13) as

$$p(\underline{y}|\underline{z}) = p(\underline{y}^p|\underline{z}^p) \ . \tag{14}$$

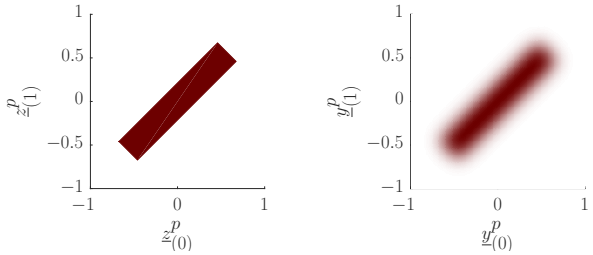
Furthermore, as the observation principle is similar, we make the assumption that we can model $p(\underline{y}^p|\underline{x})$ in the same way as $p(\underline{z}^p|\underline{x})$. Hence, we extend (11) and say that $p(\underline{y}^p|\underline{x}) = c_S$ for $\underline{y}^p \in \mathcal{S}$. Thus, we see that, by marginalizing y^t out of $p(y^t, \underline{y}^p|\underline{x})$, we obtain

$$p(\odot^-, \underline{y}^p|\underline{x}) + p(\odot^+, \underline{y}^p|\underline{x}) = p(\underline{y}^p|\underline{x}) = c_S,$$

In consequence, we only need to concern us with $p(\odot^+, \underline{y}^p|\underline{x})$, as the other likelihood for $\odot^-$ follows from

$$p(\odot^-, \underline{y}^p|\underline{x}) = c_S - p(\odot^+, \underline{y}^p|\underline{x}) \ , \tag{15}$$

and both likelihoods are 0 if $\underline{y}^p \notin \mathcal{S}$.



(a) Silhouette function $p(\odot^+|\underline{z}^p, \underline{x})$. (b) Auxiliary term $\mathcal{L}^+_{\underline{x}}(\underline{y}^p)$.

Figure 6: Left figure is the silhouette function for the shape in Fig. 2. Brown indicates a probability of 1 to obtain a positive observation at a given position, and white a probability of 0. To the right is the auxiliary term $\mathcal{L}^+_{\underline{x}}(\underline{y}^p)$, which can be seen as the result of applying position noise on Fig. 6a.

C. Deriving a Likelihood

In a similar fashion as (3), we obtain a likelihood for a received measurement $\underline{y}$ by marginalizing $\underline{z}^p$, i.e.,

$$p(\underline{y}|\underline{x}) = p(y^t, \underline{y}^p|\underline{x}) = \int_{\mathbb{R}^2} p(\underline{y}|\underline{z}) \cdot p(\underline{z}|\underline{x}) \ \mathrm{d}\underline{z}^p \ .$$

We will focus on the case for $y^t = \odot^+$. Using the simplified source probability (12) and noise probability (14), we obtain

$$p(\odot^+, \underline{y}^p|\underline{x}) = c_S \cdot \int_{\mathcal{S}} p(\underline{y}^p|\underline{z}^p) \cdot p(\odot^+|\underline{z}^p, \underline{x}) \ \mathrm{d}\underline{z}^p$$

As an intermediate step, we introduce the auxiliary term

$$\mathcal{L}_x^+(\underline{y}^p) := \int_{\mathcal{S}} p(\underline{y}^p|\underline{z}^p) \cdot p(\odot^+|\underline{z}^p, \underline{x}) \ \mathrm{d}\underline{z}^p \ . \tag{16}$$

Finally, using (15), we obtain the likelihood in terms of $\mathcal{L}_x^+(\underline{y}^p)$,

$$p(y^t, \underline{y}^p|\underline{x}) = \begin{cases} 0, & \text{if } \underline{y}^p \notin \mathcal{S} \\ c_S \cdot \mathcal{L}_x^+(\underline{y}^p) & \text{if } y^t = \odot^+ \\ c_S \cdot \left(1 - \mathcal{L}_x^+(\underline{y}^p)\right) & \text{if } y^t = \odot^-. \end{cases} \tag{17}$$

We observe that c_S is state-independent and always appears as a coefficient. Thus, an estimator can safely ignore it, so that the observed space $\mathcal{S}$ does not need to be modeled explicitly. To summarize, we see that in order to obtain a likelihood we need three steps. First, we need a silhouette function $p(\odot^+|\underline{z}^p, \underline{x})$ that describes the shape. Then, we apply sensor noise on it to produce $\mathcal{L}_x^+(\underline{y}^p)$. Finally, we plug this term into (17) to obtain the desired likelihood. In the following sections, we will derive the likelihoods for SDMs and Kernel GAMs.

D. SDMs for Silhouette Models

We are now concerned with reinterpreting the source selection probabilities from Sec. III-A in the framework of silhouette models. We observe that those terms work under the assumption that observations are always positive. Thus, for instance, the source selection from the traditional SDM in (4) can be rewritten in terms of Silhouette Models as

$$p(\underline{z}^p|\odot^+, \underline{x}) = \frac{1}{|\mathcal{Z}^x|} \cdot \mathbf{1}_{\mathcal{Z}^x}(\underline{z}^p) \ .$$

We can obtain a silhouette function for this SDM by applying Bayes' rule, in the form of

$$\begin{aligned} p(\odot^+|\underline{z}^p, \underline{x}) &= \frac{p(\odot^+|\underline{x})}{p(\underline{z}^p|\underline{x})} \ \cdot \ p(\underline{z}^p|\odot^+, \underline{x}) \\ &= \underbrace{p(\odot^+|\underline{x}) \cdot |\mathcal{S}|}_{:=A} \ \cdot \ p(\underline{z}^p|\odot^+, \underline{x}) \ . \end{aligned} \tag{18}$$

As $p(\odot^+|\underline{x})$ is the ratio of positive sources observable in $\mathcal{S}$, multiplying it by the area of $\mathcal{S}$ yields the area covered by the positive sources. Thus, it follows that $A = |\mathcal{Z}^x|$, so that

$$p(\odot^+|\underline{z}^p, \underline{x}) = \mathbf{1}_{\mathcal{Z}^x}(\underline{z}^p) \ , \tag{19}$$

which can be intuitively seen in Fig. 6a. In essence, the only change is the removal of the normalization factor. Finally, the auxiliary term from (16) has the form

$$\mathcal{L}_x^+(\underline{y}^p) = \int_{\mathcal{Z}^x} \mathcal{N}\left(\underline{y}^p - \underline{z}^p; \underline{0}, \mathbf{C}^v\right) \ \mathrm{d}\underline{z}^p \ ,$$

which then leads to the likelihood using (17). It can be seen that the requirement to obtain a silhouette function like (19) is that the area of the shape is non-zero. Thus, lower-dimensional embedded shapes, for example lines and points in $\mathbb{R}^2$, cannot be directly described using this model. In particular, this is problematic because we cannot use simple GAMs from Sec. III-B, as they assume a single source point. However, we can still take advantage of the Kernel GAMs from Sec. III-C.

E. Kernel GAMs for Silhouette Models

As in Sec. IV-D, obtaining a silhouette function from Kernel GAMs is straightforward. Thus, (8) can be rewritten as

$$p(\underline{z}^p|\odot^+, \underline{x}) = \mathcal{N}\left(\underline{z}^p - \pi_x(\underline{y}^p); \underline{0}, \mathbf{C}_x^K\right) \ , \tag{20}$$

In order to apply (18) we need to obtain the area term A for the considered kernel. A simple solution is by noting that the projection $\pi_x(\underline{y}^p)$ belongs to $\mathcal{Z}^x$, and in consequence, we expect $p(\odot^+|\underline{z}^p, \underline{x})$ to be equal to 1 at $\underline{z}^p = \pi_x(\underline{y}^p)$. Thus, from (18) and (20) we obtain

$$1 = A \cdot \mathcal{N}\left(\pi_x(\underline{y}^p) - \pi_x(\underline{y}^p); \underline{0}, \mathbf{C}_x^K\right) \ ,$$

which leads to the kernel area of

$$A = \frac{1}{\mathcal{N}\left(\underline{0}; \underline{0}, \mathbf{C}_x^K\right)} = \frac{(2\pi)^{\frac{2}{2}} \left|\mathbf{C}_x^K\right|}{\exp(\underline{0})} = 2\pi \left|\mathbf{C}_x^K\right| \ ,$$

where $|\cdot|$ denotes the determinant. Finally, we obtain

$$p(\odot^+|\underline{z}^p, \underline{x}) = 2\pi \left|\mathbf{C}_x^K\right| \cdot \mathcal{N}\left(\underline{z}^p - \pi_x(\underline{y}^p); \underline{0}, \mathbf{C}_x^K\right) \ .$$

Obtaining the auxilary term from (16) consists of a simple addition of the noise covariance matrices, in a similar fashion as (9), yielding

$$\mathcal{L}_x^+(\underline{y}^p) = 2\pi \left|\mathbf{C}_x^K\right| \cdot \mathcal{N}\left(\underline{y}^p - \pi_x(\underline{y}^p); \underline{0}, \mathbf{C}_x^K + \mathbf{C}^v\right) \ .$$

For this expression, the maximum value is $\frac{|\mathbf{C}_x^K|}{|\mathbf{C}_x^K + \mathbf{C}^v|} \leq 1$ and thus, (17) is guaranteed to be non-negative.

V. Implementation

In this section, we will describe the model implementation used in the evaluation. But first, we will present a short motivation. Assuming that only positive observations are available, SDMs generally produces excellent results [1] as they fully exploit the available information about $p(\underline{z}|\underline{x})$ (assuming it is known a priori). Thus, SDMs present an appropriate standard against which to compare the proposed models. Keeping this in mind, we will implement three approaches for tracking a rectangular shape as in Fig. 1. The first is a traditional rectangle *SDM* as explained in the example in Sec. III-A. The second is a line segment Kernel-GAM which implements Silhouette Models (*S-GAM*) as described in Sec. IV-E. Finally, the third is a rectangle SDM with Silhouette Models (*S-SDM*), as explained in Sec. IV-D. Thus, we want the evaluation to answer two questions. On the one hand, we want to know how the simple and easy-to-implement S-GAM fares against the more complex but accurate SDMs. And on the other hand, we want to find out how the S-SDM deals with occlusions given the additional information of negative observations. The advantage of a rectangle is that, in the special case of isotropic noise where $\mathbf{C}^v = \sigma_v^2 \cdot \mathbf{I}_2$, and where positive sources are

uniformly distributed on the (possibly occluded) shape, we can obtain a closed-form definite integral.

In this section, we will discuss four aspects of the implementation. First, we will define the shape to be estimated and its corresponding state. Second, we will obtain a solution for the SDM integral over the rectangle. Third, we need to define the projection function $\pi_x(\underline{y}^p)$ used by the S-GAM. Finally, we describe the Bayesian estimator used in the evaluation.

A. Defining the State

At the time step k, the state $\underline{x}$ has the form

$$\underline{x}_k = [\ \underline{c}_k, \alpha_k, l_k, h_k, \nu_k, \omega_k\ ]^T ,$$

where $\underline{c}_k$ is the center, α_k the orientation angle, and l_k the length of the shape. The term h_k represents the lateral extent of the shape. For SDMs, it is interpreted as the height of the rectangle. For the Kernel GAM, h_k is the kernel size, so that the kernel covariance matrix has the form $\mathbf{C}_x^K = h_k^2 \cdot \mathbf{I}_2$. As the target can be moving, we also estimate the translational speed ν_k and the rotational velocity ω_k. The direction of the movement is assumed to be α_k. We define the orientation matrix $\mathbf{R}(\cdot)$ as

$$\mathbf{R}(\alpha_k) := \begin{pmatrix} \cos(\alpha_k) & -\sin(\alpha_k) \\ \sin(\alpha_k) & \cos(\alpha_k) \end{pmatrix} .$$

B. Integral over a Rectangle

In order to describe the integral over a rectangle, first we define the function $G(\eta_0, \eta_1, \mu, \sigma)$, which calculates the definite integral of $\mathcal{N}(\eta; \mu; \sigma^2)$ in the range $[\eta_0, \eta_1]$, i.e.,

$$G(\eta_0, \eta_1, \mu, \sigma) := \frac{1}{2}\left(\operatorname{erf}\left(\frac{\eta_1 - \mu}{\sqrt{2}\sigma}\right) - \operatorname{erf}\left(\frac{\eta_0 - \mu}{\sqrt{2}\sigma}\right)\right) .$$

Furthermore, we define

$$\underline{\eta} = \left[\eta_{(0)}, \eta_{(1)}\right]^T := \mathbf{R}(\alpha_k)^{-1} \cdot (\underline{y}^p - \underline{c}_k) .$$

Finally, based on the results of [1], the auxiliary term $\mathcal{L}_x^+(\underline{y}^p)$ for the Silhouette SDM becomes

$$\mathcal{L}_x^+(\underline{y}^p) = G\left(-\frac{l_k}{2}, \frac{l_k}{2}, \eta_{(0)}, \sigma_v\right) \cdot G\left(-\frac{h_k}{2}, \frac{h_k}{2}, \eta_{(1)}, \sigma_v\right)$$

which together with (17) yields its likelihood. In turn, the likelihood for the traditional SDM (5) becomes

$$p(\underline{y}|\underline{x}) = \frac{1}{A} \cdot \mathcal{L}_x^+(\underline{y}^p) ,$$

where $A = l_k \cdot h_k$.

C. The Projection Function

The projection function $\pi_x(\underline{y}^p)$ is the Euclidian projection onto the line segment, which is determined by its end points

$$\underline{a}_k := \underline{c}_k - \mathbf{R}(\alpha_k) \cdot \left[\tfrac{1}{2} l_k, 0\right]^T \text{ , and}$$
$$\underline{b}_k := \underline{c}_k + \mathbf{R}(\alpha_k) \cdot \left[\tfrac{1}{2} l_k, 0\right]^T .$$

The Euclidian projection can be obtained in closed-form as

$$\pi_x(\underline{y}^p) = \underline{a}_k + \text{clamp}(u^*) \cdot (\underline{b}_k - \underline{a}_k) ,$$

where $\text{clamp}(\cdot) := \max(0, \min(\cdot, 1))$, and

$$u^* := \frac{(\underline{y}^p - \underline{a}_k)^T (\underline{b}_k - \underline{a}_k)}{(l_k)^2} .$$

D. Gaussian Bayesian Estimator

A Gaussian Bayesian estimator consists of two steps which alternate between *updating* and *predicting* the state. At the time step k, the uncertainty of the state is described by the prior $f_k^p(\underline{x}_k)$, assumed to be a Gaussian distribution in the form of $f_k^p(\underline{x}_k) = \mathcal{N}(\underline{x}_k; \hat{\underline{x}}_k, \mathbf{C}_k^x)$. In the first step, after receiving the set of measurements $\mathcal{Y}_k$, the state is *updated* using Bayes' rule, described by

$$f_k^e(\underline{x}_k) = c_B \cdot p(\mathcal{Y}_k|\underline{x}_k) \cdot f_k^p(\underline{x}_k) , \tag{21}$$

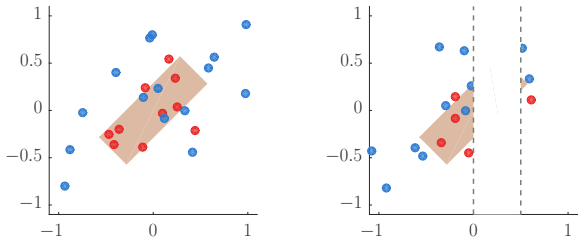
where $p(\mathcal{Y}_k|\underline{x}_k)$ is derived from (2), and c_B is a normalization constant. In the second step, the state at the next time step $k+1$ is *predicted* by

$$\underline{x}_{k+1} = a(\underline{x}_k) + \underline{r}_k , \tag{22}$$

where $\underline{r}_k$ is the Gaussian distributed process noise $\underline{r}_k \sim \mathcal{N}(\underline{0}, \mathbf{Q}_k)$. The function $a(\cdot)$ models how the state evolves in time, and for the evaluation it has the form

$$a(\underline{x}_k) = \begin{bmatrix} \underline{c}_k \\ \alpha_k \\ \vdots \end{bmatrix} + \begin{bmatrix} \mathbf{R}(\alpha_k) \cdot [\Delta t_k \cdot \nu_k, 0]^T \\ \Delta t_k \cdot \omega_k \\ \\ \mathbf{0}_{4\times 1} \end{bmatrix} ,$$

where Δt_k is the elapsed time. In other words, $a(\underline{x}_k)$ moves the shape with speed ν_k in the direction α_k, while rotating it with angular velocity ω_k. The other parameters remain intact. As neither the posterior (21) nor the prediction (22) are in general Gaussian distributed, in particular considering that $a(\underline{x}_k)$ is non-linear, we will use the *Progressive Gaussian Filter* (PGF) [17] in order to obtain suitable Gaussian approximations.



(a) No occlusion zones. (b) With occlusion zone (white stripe).

Figure 7: Shape (light brown) and measurements (positive in red, negative in blue) of the static experiment. Note that, due to sensor noise, some positive observations end up outside of the shape, and some negative observations inside.

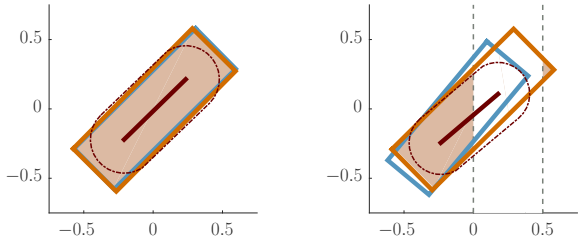
VI. Evaluation

In this section we will evaluate the three approaches presented in Sec. V. We will realize two experiments based on synthethic data, the first using a static target and the second using a moving target, and considering two cases, without and with occlusions. The target is a rectangle of size $0.2\,\text{m} \times 0.8\,\text{m}$, seen in (7). The measurement generation is as follows. First, 10 sources are generated from inside the target, and 20 sources from outside. We believe this ratio is reasonable, as sensors typically scan a large space and can produce very large amounts of negative observations. The negative sources are drawn uniformly from a rectangular envelope of $0.4\,\text{m} \times 1.2\,\text{m}$, centered on the target and with the same orientation. Second, the positions of these sources are corrupted with noise according

to (1), where $\mathbf{C}^v = \sigma_v^2 \cdot \mathbf{I}_2$ and $\sigma_v = 0.1\,\text{m}$. Finally, all measurements inside *occlusion zones* (e.g., the white stripe in Fig. 7b) are dropped. Note that we differentiate between negative observations, which tell us where the target is not, and occluded regions, where there are no measurements.

A. Static Target

First, we will realize a simple experiment where the target is static, as seen in Fig. 7. The initial state estimate is $f_k^p(\underline{x}_0) = \mathcal{N}(\underline{x}_0; \hat{\underline{x}}_0, \mathbf{C}_0^x)$. For $\hat{\underline{x}}_0$, the center is $\underline{c}_0 = [-1, 0]$, the angle is $\alpha_0 = 0$, the length is $l_0 = 0.2$, and the height parameter is $h_0 = 0.1$. The remaining parameters are 0. Finally, $\mathbf{C}_0^x = 10^{-3} \cdot \mathbf{I}_7$. At each time step, we assume a process noise of $\mathbf{Q}_k = 10^{-2} \cdot \mathbf{I}_7$. The occlusion zone, when active, drops all measurements inside the rectangle $[0, 0.5] \times [-1.5, 1.5]$. Note that a small part of the target remains visible to the right of the occlusion zone.



(a) No occlusion zones, $k = 20$. (b) With occlusion zone, $k = 100$.

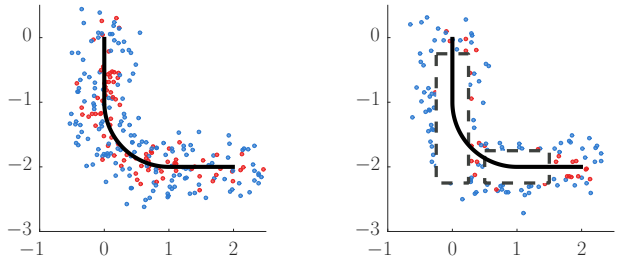
Figure 8: Results of the static experiment. SDM in blue, S-GAM in dark brown, and S-SDM in orange.

Fig. 8 shows the average result at the given time step after 100 runs. The dotted line around the S-GAM line serves simply to illustrate the kernel size, and contains all points whose distance to the line is h_k. For the case without occlusions (Fig. 8a), all approaches quickly found the target by time step $k = 20$. Both SDMs, as expected, had no trouble describing the rectangle. In addition, we confirm that the S-GAM, unlike traditional GAMs, has no trouble with overestimating the extent. The S-GAM was able to determine the center and the orientation, but we note that the length is slighty lower than the ground truth. For the case with occlusions (Fig. 8b), the expected distribution of sources ceases to be valid. In consequence, the traditional SDM is still stuck in an incorrect position by $k = 100$. The S-GAM has the right position, but is slightly biased in the angle, and the length is noticeably shorter. The S-SDM, however, had no trouble finding the ground truth, even considering the large occlusion. This reflects the fundamental difference in how both SDMs interpret the observations. The traditional SDM inteprets a lack of observations as the object not being there, while the S-SDM does not make any assumption from the lack of observations.

B. Moving Target

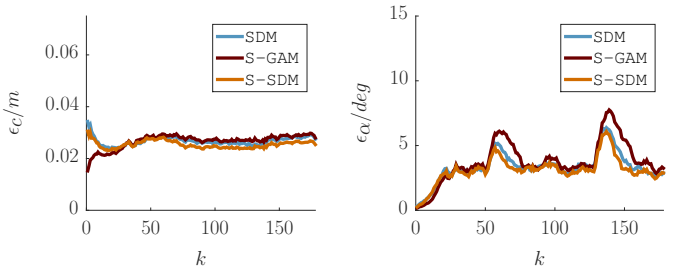
A more detailed evaluation follows from the tracking of a moving target. In this case, the ground truth rectangle starts in a vertical position at $[0, 0]$, and moves along the black path in Fig. 9a, so that its orientation is always tangential to the path. The target moves $0.02\,\text{m}$ each time step, and the path has a length of $3.5\,\text{m}$, for a total of 178 time steps. If active, two rectangular occlusion zones appear on the path (Fig. 9b) at $[-0.25, 0.25] \times [0.25, 2.25]$ and $[0.5, 1.5] \times [1.75, 2.25]$. The initial state uncertainty is $f_k^p(\underline{x}_0) = \mathcal{N}(\underline{x}_0; \hat{\underline{x}}_0, \mathbf{C}_0^x)$. For $\hat{\underline{x}}_0$, the centers, angles, and velocities were initialized with the correct values, as we are more interested in how the estimator keeps up with the target. The length is set to $l_0 = 0.2$, and the height parameter is $h_0 = 0.1$. Finally, $\mathbf{C}_0^x = 10^{-3} \cdot \mathbf{I}_7$. At each time step, we assume a process noise of $\mathbf{Q}_k = \text{diag}(0, 0, 0, 10^{-4}, 10^{-4}, 10^{-5}, 10^{-2})$.

For the case without occlusions, the results are very similar for the three models, both in the center error (Fig. 10a) and angle error (Fig. 10b). We see that all could keep up with the target mostly without any problem. Thus, we can validate that the S-GAM can track the pose of a target in a comparable way to traditional SDMs, even if the extent is slightly biased. However, the situation changes for the case with occlusions, as the occlusion zones block most of the positive observations. The results for four representative time steps can be seen in Fig. 11. On the one hand, the traditional SDM becomes unstable while the target is occluded, as can be seen from the RMSE (Fig. 12) after 100 runs. On the other hand, the Silhouette Models produce similar results to the unoccluded scenario. This indicates that S-GAM and S-SDM can track the target even they receive little to no positive observations. In other words, they can navigate blindly yet successfully using only information of where the target is *not*.



(a) No occlusion zones. (b) Two occlusion zones.

Figure 9: Path traversed by the moving target (black), with occlusion zones (gray) and representative measurements.

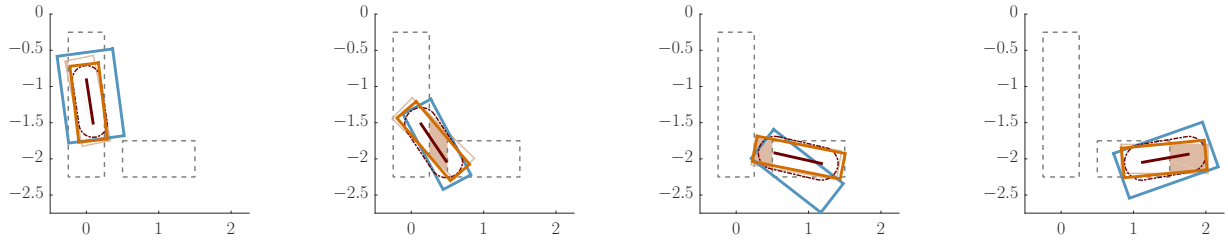


(a) Center RMSE. (b) Angle RMSE.

Figure 10: Results for the moving target without occlusion.

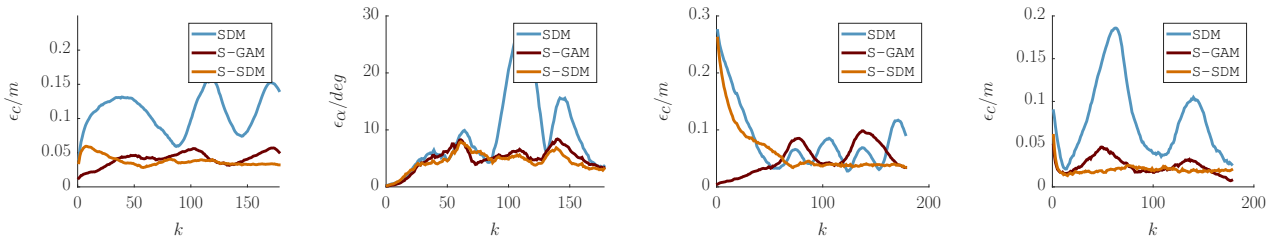
VII. Conclusion

When tracking extended objects, we usually receive several measurements of which we know, with a given certainty, that they do not belong to the target. Traditionally, these measurements are discarded, although they contain valuable information about where the target is not. In this paper, we introduced Silhouette Models, that aim to exploit this additional



(a) Results at $k = 60$. (b) Results at $k = 90$. (c) Results at $k = 120$. (d) Results at $k = 150$.

Figure 11: Target estimates. Visible parts of the shape in light brown, SDM in blue, S-GAM in dark brown, and S-SDM in orange. Parts the target within the occlusion zones (white rectangles) are hidden from view and do not produce any measurements.



(a) Center RMSE. (b) Angle RMSE. (c) Length RMSE. (d) Width RMSE.

Figure 12: Root Mean Square Errors for the center, angle, length and width.

information. First, we presented two traditional models that work only with positive observations. On the one hand, SDMs exploit information about the source distribution, but their likelihoods are usually difficult to evaluate, and do not work well under occlusions. On the other hand, GAMs are easy to implement and fast to evaluate, but they have problems with estimating the extent. In order to see whether negative observations could solve their weaknesses, we extended both approaches with Silhouette Models. We were left with two questions. On the one hand, we wanted to know how the extended GAMs fared against SDMs. We saw that the accuracy in pose estimation was very similar, and that negative observations solved the extent issue, making the extended GAM a versatile yet extremely fast tracking approach. On the other hand, we wanted to find out whether extending an SDM allows it to deal better with occlusions. In this aspect, we showed that both Silhouette Models could easily track a target even with almost no positive observations, by using only information of where the target is not.

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] M. Baum, B. Noack, and U. D. Hanebeck, "Extended Object and Group Tracking with Elliptic Random Hypersurface Models," in *Proceedings of the 13th International Conference on Information Fusion (Fusion 2010)*, Edinburgh, United Kingdom, Jul. 2010.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

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

[9] 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.

[10] 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, Jul. 2012.

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

[12] 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.

[13] 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.

[14] F. Faion, M. Baum, and U. D. Hanebeck, "Silhouette Measurements for Bayesian Object Tracking in Noisy Point Clouds," in *Proceedings of the 16th International Conference on Information Fusion (Fusion 2013)*, Istanbul, Turkey, Jul. 2013.

[15] A. Laurentini, "The visual hull concept for silhouette-based image understanding," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 16, no. 2, pp. 150–162, Feb 1994.

[16] K. Grauman, G. Shakhnarovich, and T. Darrell, "A bayesian approach to image-based visual hull reconstruction," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 1, June 2003, pp. I–187–I–194 vol.1.

[17] 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.