Fusion of Elliptical Extended Object Estimates Parameterized with Orientation and Axes Lengths

Kolja Thormann [1,1,1] and Marcus Baum [2]

[1]University of Goettingen
[2]Affiliation not available

November 8, 2023

Abstract

This article considers the fusion of target estimates stemming from multiple sensors, where the spatial extent of the targets is incorporated. The target estimates are represented as ellipses parameterized with center orientation and semi-axis lengths and width. Here, the fusion faces challenges such as ambiguous parameterization and an unclear meaning of the Euclidean distance between such estimates. We introduce a novel Bayesian framework for random ellipses based on the concept of a Minimum Mean Gaussian Wasserstein (MMGW) estimator. The MMGW estimate is optimal with respect to the Gaussian Wasserstein (GW) distance, which is a suitable distance metric for ellipses. We develop practical algorithms to approximate the MMGW estimate of the fusion result. The key idea is to approximate the GW distance with a modified version of the Square Root (SR) distance. By this means, optimal estimation and fusion can be performed based on the square root of the elliptic shape matrices. We analyze different implementations using, e.g., Monte Carlo methods, and evaluate them in simulated scenarios. An extensive comparison with state-of-the-art methods highlights the benefits of estimators tailored to the Gaussian Wasserstein distances.

Fusion of Elliptical Extended Object Estimates Parameterized with Orientation and Axes Lengths

Kolja Thormann and Marcus Baum

***Abstract*—This article considers the tracking of elliptical extended targets parameterized by center, orientation, and semi-axes. The focus of this article lies on the fusion of extended target estimates, e.g., from multiple sensors, by handling the challenges introduced due to ambiguities in this parameterization and the unclear meaning of the Mean Square Error (MSE). For this purpose, we introduce a novel Bayesian framework for elliptic extent estimation and fusion based on two new concepts: (1) a probability density function for ellipses called Random Ellipse Density (RED) which incorporates the ambiguities that come with the ellipse parameterization, and (2) the Minimum Mean Gaussian Wasserstein (MMGW) estimate which is optimal with respect to the squared Gaussian Wasserstein (GW) distance – a suitable distance metric on ellipses. We develop practical algorithms for ellipse fusion and approximating the MMGW estimate. Different implementations, e.g., based on Monte Carlo simulation, are introduced and compared to state-of-the-art methods, highlighting the benefits of estimators tailored to the GW distance.**

***Keywords*— Extended Object Tracking, Information Fusion, Bayesian Estimation**

I. Introduction

In many modern tracking applications the resolution of the involved sensors is high enough to resolve the spatial extent of the targets. For this reason, Extended Object Tracking (EOT) methods that estimate both the shape and kinematic parameters of a target are becoming increasingly important [1], [2]. Most EOT methods have been developed for sensors that resolve a varying number of noisy Cartesian detections from the target, e.g., based on a spatial distribution model [3]. The extent can be modeled by basic shapes like rectangles [4], [5] or ellipses [2], [6]–[9]. More detailed shapes are also possible by modelling them as a combination of multiple random matrices [10] or using a Random Hypersurface Model (RHM). The latter describes star-convex shapes and was modeled by, e.g., Fourier coefficients [11], Gaussian processes [12]–[14], or splines [15], [16].

To improve tracking, multiple sensors can be applied. Communications can be reduced by sending only compact information, i.e., by tracking the target locally and transmitting only the estimate and covariance instead of the individual point clouds (given, e.g., a radar sensor) [17]. These estimates can be shared between neighboring nodes in a sensor network [18] or sent to a central fusion unit [17]. In this work, we consider object level fusion in a centralized fusion unit, i.e., we want to fuse the local state densities and determine a global estimate. Thus, it is essential that the sensors do not only provide their estimates, but their uncertainties as well [19].

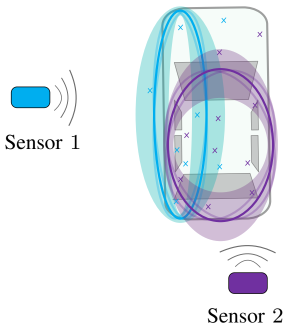


Fig. 1: Ground truth of a vehicle along with the internal measurement point clouds of sensors 1 and 2 in blue and purple respectively, as well as their estimates as ellipses. The transparent ellipses represent orientation uncertainty and the transparent areas visualize the uncertainty in the semi-axes.

Elliptic shapes are widely-used to approximate the target's extent [2], [6]–[9]. An example application is automotive radar, as the detections are usually scattered across the entire surface. Elliptic shapes are useful for targets such as pedestrians [20], but can also be applied in scenarios with high sensor noise for other objects, if their actual form is hard to estimate (see, e.g., the radar point clouds from the nuScenes data set [21]). Another example is marine radar, as again the detections are scattered across the surface while the targets themselves, usually ships, are also elliptical (see, e.g., the Data Association with Aids to Navigation (DAAN) dataset [22]). Furthermore, group targets such as groups of pedestrians can be approximated by ellipses as well, e.g., [2], [23].

Elliptic targets can be tracked using Random Matrices, pioneered by Koch et al. [6], [24]. However, it was demonstrated in [7] that under the assumption that the semi-axes don't change, but the orientation does, tracking the orientation and semi-axes as individual parameters can improve the estimation, especially in turns. Also, if the same target is detected by different sensors, the estimates may have different uncertainties depending on the sensor's qualities and position relative to the target (see Figure 1). Under these conditions, there might be different uncertainties for the semi-axes or an increased uncertainty about the target's orientation due to, e.g., high noise or maneuvers like driving around a corner. As a result, the densities must be fused, not only their local estimates, to get a proper global estimate.

The goal now is to fuse the estimates and covariances

Kolja Thormann and Marcus Baum are with the Institute of Computer Science, University of Goettingen, Germany, {kolja.thormann, marcus.baum}@cs.uni-goettingen.de

provided by the sensors appropriately, preserving as much information about the uncertainties of the different parameters as possible. However, when parameterizing the ellipse with orientation and semi-axes, new unique challenges occur such as ambiguity, i.e., the same ellipse being parameterized in different ways, as a result of different initialization or tracking in high process and sensor noise. To counter this, the shape orientation could be aligned with the velocity vector. However, in tracking elliptic targets, velocity and shape orientation are often decoupled, e.g., [6], [7]. The reason is that the ellipse orientation is not necessarily equal to the orientation of the velocity vector, e.g., due to drifting of the vessels in case of ships [22], [25]. Moreover, if a target is stationary, there is no velocity vector. Another way to handle these ambiguities for the estimates at least is by simply transforming them to shape matrices, but fusing the entire state densities is not so trivial. This article provides a method to fuse state densities under these challenges and to find an appropriate estimate incorporating the geometric properties of the estimated shape.

A. Contributions

The main contribution of this article is a novel Bayesian framework for estimation and (shape) fusion with elliptic extended targets parameterized by center, width, length, and orientation. The contributions include

- the promotion of the squared Gaussian-Wasserstein (GW) distance [26] as a risk function on elliptic shapes, defining a Minimum Mean Gaussian Wasserstein (MMGW) estimator,
- the introduction of a suitable probability density function on the explicit extent parameters of ellipses, a Random Ellipse Density (RED),
- the derivation of a MMGW estimator approximation and fusion method by replacing the GW distance via an extension of the Square Root (SR) distance [27], the Extended Square Root (ESR) distance,
- the development of practical implementations of the MMGW and RED concepts, and
- a comparison of the MMGW estimator and the fusion via REDs with state-of-the-art concepts, demonstrating improved performance in high noise scenarios.

This article is based on our previous conference publication [28]. We further develop our previous results by

- introducing the concept of RED, which allows for a meaningful Bayesian fusion of uncertain ellipses, along with an efficient mixture reduction as an improved fusion method to the one presented previously,
- providing more insight on the effectiveness of approximating the GW distance with the ESR distance, and
- presenting a new and much more elaborate experimental evaluation of the fusion methods using moving targets.

B. Related Work

As elliptic targets are often represented by random matrices [6], [24], fusion methods for this representation can be found in literature as well. These include a combination of two random matrix estimates utilizing their respective Poisson rates [29], with a focus on the combination of targets, also considering that their size might increase. In [18], the authors argue that as Random Matrices are represented by Inverse-Wishart distributions, the weighted average of the shape matrices (and of the degree of freedom) respectively minimizes the Kullback-Leibler divergence to the densities. Another approach using point clouds from different sensors directly, including a method applying a particle filter, can be found in [30]. The latter draws its particles from an importance distribution in the space of the explicit parameters with the orientation, length, and width of the shape matrix as mean. It then uses the random matrix likelihood to weight the particles with the point clouds generated by the sensors. There is an extension of this method for asynchronous sensors in [31], creating local estimates as particle densities, approximating them as Gaussian mixtures, and then fusing them via geometric mean densities. In both works, the mean of the particle density is calculated as the weighted average of the shape matrices. Another recent work [32] presents a distributed fusion method based on a variational Bayesian approach. They model the state also as a random matrix, define latent variables as noise-free measurements, and distribute their statistics across the sensors to approximate the state globally. [33] propose a combination strategy of ellipses parameterized with orientation and semi-axes in a distributed sensor network based on the diffusion Kalman filter. They do however assume that point clouds are shared between neighboring sensors. Additionally, they combine the neighboring estimates after the measurement update by turning them into shape matrices and finding combination weights. They do however not consider the covariances in the fusion, which would be able to provide additional information and improve the estimation [34].

For rectangles, [35] provide a method which focuses on associating and fusing rectangular shapes by using the covariances of their corners. In [36], a method is presented to fuse rectangular estimates modeled by center, orientation, length, and width in a Kalman fashion, but considering one of the corners as a reference point. An approach to fuse only segments, represented by points, lines, or L-shapes, is introduced in [37].

For arbitrary shapes there exists a framework by [19] to combine star-convex forms represented by Gaussian processes, again using measurements directly. There is also an extension in [38] which fuses star-convex shapes by determining a new center and then a new radial function based on the input radial functions relative to the new center. In [39], estimated Random Finite Sets (RFS) from multiple sensors are combined using the Kullback-Leibler Divergence between them.

To the best of our knowledge, there exist no fusion methods for ellipses explicitly parameterized by orientation and semi-axes in literature which go beyond applying a Kalman filter on the state densities or simply combining estimates without considering their covariances [33]. Additionally, we know of no other approach in tracking literature to explicitly change the risk function in estimating an extended target based on its posterior density. This idea is inspired by the Minimum Mean Optimal Subpattern Assignment (MMOSPA) estimators [40],

[41], which estimate multiple point target densities [42] by minimizing the Optimal Subpattern Assignment (OSPA) distance [43]. MMOSPA estimation [41] (and this work) is also related to the concept of a Wasserstein Barycenter [44]–[47].

C. Structure

The remainder of the paper is organized as follows. The problem this paper deals with is described in Section II. Then, a novel fusion method for elliptical extended targets based on REDs is introduced in Section III, along with a more insightful recapitulation of the MMGW estimator. This is followed by approximations and implementations of fusion methods based on the newly derived RED and MMGW estimator in Section IV. Next, Section V provides an evaluation of the estimators and the fusion methods and then the results are discussed in Section VI. This article is concluded in Section VII.

II. Problem formulation

This work considers the fusion of elliptic target estimates stemming from multiple sources, e.g., sensors tracking the same target locally and then sending their estimate to a fusion center. We assume the sources provide the estimate along with its uncertainty as a covariance matrix. To clarify the setting, we first describe the state we want to estimate in Section II-A and the fusion model we consider in Section II-B. In the following, lower case letters denote scalars, lower case bold letter denote vectors, and upper case bold letters denote matrices.

A. State

We model the spatial extent by a center $\mathbf{m} = \begin{bmatrix} m_1 & m_2 \end{bmatrix}^{\mathrm{T}}$, an orientation α, and semi-axis length l and width w (see also Figure 2a). This representation explicitly allows for capturing uncertainties for the different dimensions, providing important information which can be utilized for the fusion. Its usefulness has been demonstrated in, e.g., [7].

We also add a kinematic state $\mathbf{r}$, which can be realized as, e.g., a Cartesian velocity $\dot{\mathbf{m}}$, a polar velocity consisting of velocity v and orientation ψ, or even only v while using α as the orientation, in either case resulting in a state vector

$$\mathbf{x} = \begin{bmatrix} \mathbf{m}_{\mathbf{x}}^{\mathrm{T}} & \alpha_{\mathbf{x}} & l_{\mathbf{x}} & w_{\mathbf{x}} & \mathbf{r}_{\mathbf{x}}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \ . \tag{1}$$

We assume the state to be a Gaussian distributed probability density with $p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}, \mathbf{C})$.

B. Fusion model

For simplicity, we assume only a single target which is tracked by multiple sensors. Each sensor can observe the entire target's shape, i.e., we assume that the sensors observe a point cloud distributed across the entire target's surface [2], [6], [7], [9], [11], [24], [48], [49]. Due to aspects such as different sensor quality or sensor to target geometry, the sensor estimates have different uncertainties, represented by their covariances.

We now assume that a fusion center keeps track of the global target state parameterized as in (1) with mean $\hat{\mathbf{x}}$ and covariance $\mathbf{C}$. A sensor with index s then provides a local estimate $\hat{\mathbf{x}}_s$ with uncertainty $\mathbf{C}_s$. They can be related to each other with the measurement equation

$$\mathbf{x}_s = \mathbf{H}\mathbf{x} + \mathbf{w}_s \ , \tag{2}$$

with $\mathbf{w}_s \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_s)$. We assume the entire shape state is provided by the sensor. If all of $\mathbf{r}$ is provided as well, we have $\mathbf{H} = \mathbf{I}_d$ with $\mathbf{I}_d$ being the d-dimensional identity matrix and d the dimension of $\mathbf{x}$. If only part or none of $\mathbf{r}$ is given by the sensor, the corresponding rows of $\mathbf{H}$ are cut. Essentially, the challenge is to effectively combine elliptic shape estimates, making use of their covariances and resulting in a new estimate $\hat{\mathbf{x}}^+$ along with a covariance $\mathbf{C}^+$.

We want to highlight the importance of tracking the individual axes and orientation along with their covariances instead of using a random matrix again, as one sensor from behind the target might give a precise estimate of the width, but has almost no information about the length, so that this parameter should be weighted less when fusing. An alternative example would be scenarios with highly noisy measurements across the entire target surface from, e.g., radar. Due to the radial noise, the uncertainty orthogonal to the measurement direction can be higher than in measurement direction.

For simplicity's sake, we assume $\mathbf{x}$ and $\mathbf{w}_s$ are independent, as the focus of this work are the unique challenges in fusing elliptic shape estimates. In practice, possible unknown correlation can be dealt with by methods such as covariance intersection [50], [51]. If enough knowledge about the system is at hand, the correlation can be tracked [17], [52].

III. Bayesian Fusion and Estimation

Consider the global ellipse estimate $\mathbf{x}$ with mean $\hat{\mathbf{x}}$ and covariance $\mathbf{C}$ and a sensor s providing a local estimate of the ellipse $\hat{\mathbf{x}}_s$ with sensor noise $\mathbf{C}_s$. The naive approach to combine them would be a linear fusion according to the Kalman filter. It finds the Minimum Mean Square Error (MMSE) estimate using the squared Euclidean distance as risk function according to

$$\hat{\mathbf{z}} = \underset{\mathbf{z}}{\operatorname{argmin}} \int ||\mathbf{z} - \mathbf{x}||_2^2 \cdot p(\mathbf{x}|\hat{\mathbf{x}}_s) \mathrm{d}\mathbf{x} = \mathrm{E}[\mathbf{x}|\hat{\mathbf{x}}_s] \ . \tag{3}$$

The fusion is conducted as

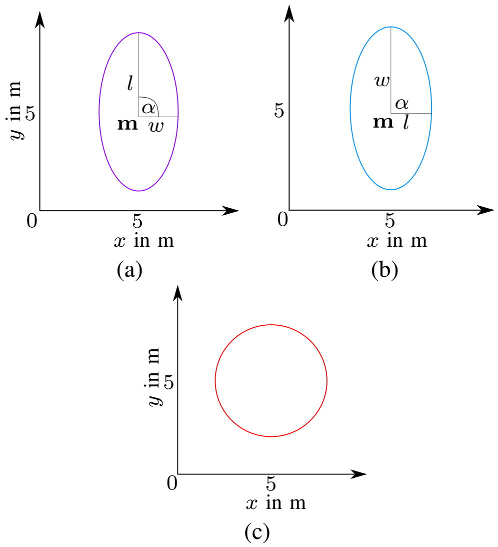
$$p(\mathbf{x}|\hat{\mathbf{x}}_s) \sim p(\hat{\mathbf{x}}_s|\mathbf{x}) \cdot p(\mathbf{x}) \ , \tag{4}$$

with prior $p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}, \mathbf{C})$ and Gaussian likelihood $p(\hat{\mathbf{x}}_s|\mathbf{x}) = \mathcal{N}(\hat{\mathbf{x}}_s; \mathbf{x}, \mathbf{C}_s)$.

To demonstrate the issues by means of an example, we consider the case of equal covariances. The estimate is then gained by averaging the two means

$$\hat{\mathbf{z}} = \frac{1}{2}(\hat{\mathbf{x}} + \hat{\mathbf{x}}_s) \ . \tag{5}$$

However, as Figure 2 shows, this approach can produce counter-intuitive results due to ambiguities in the parameterization. The two ellipses in Figures 2a and 2b are geometrically the same, just with an orientation shift of $\frac{\pi}{2}$ and the semi-axes switched, resulting in a fusion in Figure 2c which averages the intuitively wrong axes ($l_{\mathbf{x}}$ with l_s and $w_{\mathbf{x}}$ with w_s in Figures 2a



(a)

(b)

(c)

Fig. 2: Ellipses with parameters $\mathbf{m_x} = [5 \quad 5]^{\mathrm{T}}$, $\alpha_{\mathbf{x}} = \frac{\pi}{2}$, $l_{\mathbf{x}} = 4$, and $w_{\mathbf{x}} = 2$ in Figure 2a and $\mathbf{m}_s = [5 \quad 5]^{\mathrm{T}}$, $\alpha_s = 0$, $l_s = 2$, and $w_s = 4$ in Figure 2b and their RMSE estimate using Euclidean distance in Figure 2c.

and 2b). Such ambiguities can occur for example due to the placement of the sensors relative to the target, e.g., if the sensors initialize a target always with the orientation facing away from the sensor. It could also happen during tracking under high process and sensor noise that different sensors end up estimating the target with different orientations.

The reason for this counter-intuitive behavior lies in Equation (3), namely from

(1) the fusion $p(\mathbf{x}|\hat{\mathbf{x}}_s)$ by means of Equation (4) and
(2) the squared Euclidean distance $||\mathbf{z}-\mathbf{x}||_2^2$ as risk function.

Both (1) and (2) deal with real vectors and do not account for the elliptic shapes. A straightforward solution is to transform the estimates into shape matrices and fuse them as done in [33]. However, to improve the fused estimate, the covariances must be considered as well. For example, our estimates can have different uncertainties in, e.g., the semi-axes, which would be lost if we only transform the estimates into shape matrices and combine them. Additionally, we want to fuse the state densities, i.e., we want to calculate the posterior distribution of the elliptic shape state to incorporate estimates consecutively. For this reason, in this work, we propose

(1) a novel probability density for elliptic shapes in Section III-A, which allows for a sound Bayesian ellipse fusion and
(2) to use the squared GW distance to define an MMGW estimator to obtain intuitive point estimates in Section III-B.

The two methods will be summarized and related in Section III-C. As the focus is on the shape state, we will omit the kinematic part $\mathbf{r}$ in this section for readability.

A. Random Ellipse Densities

To define a density for ellipses, we need to deal with the ambiguities and the constraint that semi-axes need to be positive. For the latter, we create a truncated normal distribution by setting a lower bound of 0 for l and w,

$$p_t(\mathbf{x}) = \begin{cases} 0 & \text{if } l < 0 \vee w < 0, \\ c \cdot p(\mathbf{x}) & \text{else}, \end{cases} \tag{6}$$

with normalizing constant c. Next, to avoid the issue visualized in Figure 2, we adapt the concept of wrapped distributions [53]. We define a transformation of $\mathbf{x}$ to represent all equivalent ellipses by

$$K_k(\mathbf{x}) = \begin{bmatrix}\mathbf{m}_{\mathbf{x}}^{\mathrm{T}} & \alpha_{\mathbf{x},k} & v_k(l_{\mathbf{x}}, w_{\mathbf{x}}) & v_{k+1}(l_{\mathbf{x}}, w_{\mathbf{x}})\end{bmatrix}^{\mathrm{T}} \quad , \tag{7}$$

with

$$v_k(l, w) = \begin{cases} l & \text{if } k \text{ is even}, \\ w & \text{if } k \text{ is odd}, \end{cases} \tag{8}$$

$$\alpha_{\mathbf{x},k} = \alpha_{\mathbf{x}} + k \cdot \frac{\pi}{2} \quad , \tag{9}$$

and $k \in \mathbb{Z}$. It is apparent that the orientation can be restricted between 0 and $\frac{\pi}{2}$ if the semi-axes are switched for each shift. As the density $p_t(\mathbf{x})$ is defined over $\alpha \in \mathbb{R}$, all shifts from $k = -\infty$ to $k = \infty$ need to be considered to include the entire density, i.e., we want to combine all input states corresponding to the same elliptic shape. Therefore, we define the wrapped distribution as

$$\tilde{p}(\mathbf{x}) = \begin{cases} \sum_{k=-\infty}^{\infty} p_t(K_k(\mathbf{x})) & \text{if } 0 \leq \alpha_{\mathbf{x}} < \frac{\pi}{2}, \\ 0 & \text{else}. \end{cases} \tag{10}$$

We call this density a Random Ellipse Density (RED). It handles the ambiguities by representing only unique ellipses. The only exception is the case of equal semi-axes, as there is no unique representation of a circle due to the angle being chosen arbitrarily. From a mathematical point of view, a circle is a zero-probability event and does not need special consideration. However, circular point masses might be of interest, because from a practical point of view, ellipses which are close to a circle should be seen as close to each other even if their angular difference is high.

It is important to note that the Euclidean mean of this multi-modal density has no meaning and would result in a similar problem as depicted in Figure 2. To get an intuitive point estimate from the RED, the MMGW estimator described in the next section can be applied.

B. MMGW Estimator

As the Euclidean distance does not handle ambiguities in the ellipse parameterization, we suggest to use a more suitable metric. In [26], potential distance metrics on ellipses are evaluated, including Intersection-over-Union [4], the GW distance [54], Kullback-Leibler Divergence, and the Hausdorff distance, e.g., [55]. They conclude that the GW distance is the most suitable measure, as it provides a single, intuitive scalar value and can be solved in closed form. Therefore, we decided to use it as a quality measure.

The MMGW estimator changes the squared Euclidean distance on the explicit parameters in (3) with a true distance on ellipses. In the following, we will provide a more detailed explanation of this estimator and its approximation via the SR distance [27].

The squared GW distance [54] is defined as

$$\begin{aligned}&\mathrm{GW}(\mathbf{m_z}, \mathbf{Z}, \mathbf{m_x}, \mathbf{X})^2\\ &\quad= ||\mathbf{m_z} - \mathbf{m_x}||_2^2 + \mathrm{Tr}[\mathbf{Z} + \mathbf{X} - 2(\mathbf{Z}^{\frac{1}{2}}\mathbf{X}\mathbf{Z}^{\frac{1}{2}})^{\frac{1}{2}}] \ . \end{aligned} \tag{11}$$

Normally used for calculating a distance between Gaussian densities, we utilize it here to calculate the distance between two ellipses. The estimate and each realization of our state density describe an ellipse, which can be transformed into a shape matrix to be used in the GW distance as follows

$$\mathbf{Z} = \mathbf{R}_{\alpha_\mathbf{z}} \cdot \begin{bmatrix} l_\mathbf{z}^2 & 0 \\ 0 & w_\mathbf{z}^2 \end{bmatrix} \cdot \mathbf{R}_{\alpha_\mathbf{z}}^\mathrm{T} \ , \tag{12}$$

$$\mathbf{R}_\alpha = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \ , \tag{13}$$

and $\mathbf{X}$ analogous. Replacing the Euclidean distance results in the new mean estimate

$$\hat{\mathbf{z}} = \underset{\mathbf{z}}{\mathrm{argmin}} \int \mathrm{GW}(\mathbf{m_z}, \mathbf{Z}, \mathbf{m_x}, \mathbf{X})^2 \cdot \tilde{p}(\mathbf{x}) \mathrm{d}\mathbf{x} \ . \tag{14}$$

This gives us a MMGW estimator. Note that this estimator is also applicable on the original density $p(\mathbf{x})$. The question is how to calculate the estimate. Calculating the Wasserstein Barycenter from samples already requires iterative optimization [56]. To obtain a closed-form solution, we utilize the Square Root (SR) distance [27]

$$\mathrm{SR}(\mathbf{Z}, \mathbf{X}) = ||\mathbf{Z}^{\frac{1}{2}} - \mathbf{X}^{\frac{1}{2}}||_{Frobenius} \ , \tag{15}$$

and extend its squared form by including the center of the ellipse, creating the squared Extended Square Root (ESR) distance

$$\begin{aligned}&\mathrm{ESR}(\mathbf{m_z}, \mathbf{Z}, \mathbf{m_x}, \mathbf{X})^2\\ &\quad= ||\mathbf{m_z} - \mathbf{m_x}||_2^2 + \mathrm{SR}(\mathbf{Z}, \mathbf{X})^2 \ . \end{aligned} \tag{16}$$

We then approximate the GW distance via the ESR distance

$$\begin{aligned}&\mathrm{GW}(\mathbf{m_z}, \mathbf{Z}, \mathbf{m_x}, \mathbf{X})^2\\ &\quad\approx ||\mathbf{m_z} - \mathbf{m_x}||_2^2 + \mathrm{Tr}[(\mathbf{Z}^{\frac{1}{2}} - \mathbf{X}^{\frac{1}{2}})(\mathbf{Z}^{\frac{1}{2}} - \mathbf{X}^{\frac{1}{2}})]\\ &\quad= ||\mathbf{m_z} - \mathbf{m_x}||_2^2 + ||\mathbf{Z}^{\frac{1}{2}} - \mathbf{X}^{\frac{1}{2}}||_{Frobenius}^2\\ &\quad= \mathrm{ESR}(\mathbf{m_z}, \mathbf{Z}, \mathbf{m_x}, \mathbf{X})^2 \ , \end{aligned} \tag{17}$$

creating an approximated MMGW estimator. We justify the approximation in Appendix A and show that it is exact if the shape matrices would commute. The advantage of this approximation is that the MMGW estimate can be determined via averaging. This means if we define the transformation

$$T(\mathbf{x}) = \begin{bmatrix} \mathbf{m_x}^\mathrm{T} & s_\mathbf{x}^{(11)} & s_\mathbf{x}^{(12)} & s_\mathbf{x}^{(22)} \end{bmatrix}^\mathrm{T} \ , \tag{18}$$

with $s^{(nm)}$ being cells of the symmetric square root matrix

$$\mathbf{R}_{\alpha_\mathbf{x}} \cdot \begin{bmatrix} l_\mathbf{x} & 0 \\ 0 & w_\mathbf{x} \end{bmatrix} \cdot \mathbf{R}_{\alpha_\mathbf{x}}^\mathrm{T} = \begin{bmatrix} s_\mathbf{x}^{(11)} & s_\mathbf{x}^{(12)} \\ s_\mathbf{x}^{(21)} & s_\mathbf{x}^{(22)} \end{bmatrix} = \mathbf{X}^{\frac{1}{2}} \ , \tag{19}$$

we get

$$\begin{aligned}\hat{\mathbf{z}} &= \underset{\mathbf{z}}{\mathrm{argmin}} \int \mathrm{GW}(\mathbf{m_z}, \mathbf{Z}, \mathbf{m_x}, \mathbf{X})^2 \cdot \tilde{p}(\mathbf{x}) \mathrm{d}\mathbf{x}\\ &\approx \underset{\mathbf{z}}{\mathrm{argmin}} \int ||T(\mathbf{z}) - T(\mathbf{x})||_2^2 \cdot \tilde{p}(\mathbf{x}) \mathrm{d}\mathbf{x}\\ &= T^{-1}(\mathrm{E}[T(\mathbf{x})]) \ , \end{aligned} \tag{20}$$

using the law of the unconscious statistician. To also include kinematic parts, the ESR distance in (16) and by that extension the transformation in (18) need to be extended by $\mathbf{r}$ analogous to $\mathbf{m}$.

Note that representing the elements of the square-root shape matrix as a state vector has been done previously using the Cholesky decomposition [11], [49]. But in our case, we represent the original ellipse parameters as a Gaussian distributed state vector and then transform the density. We use matrix square root instead of Cholesky decomposition as it is the type of square root also used in the calculation of the GW distance and is therefore necessary in our formulas to minimize the squared GW distance. Additionally, as we have the original ellipse parameters, calculation of the square root as in (19) is straightforward.

C. Summary

To summarize our method and to give an overview of the spaces and transformations we consider, we refer to Figure 3. We assume estimates are given in ellipse parameter space, consisting of center, orientation, semi-axes lengths, and kinematics. We further assume the estimates to be regular densities, i.e., the parameters are Gaussian distributed. We referred to two challenges we want to tackle in this article, fusion and estimation.

To handle the challenges introduced by ambiguity in the fusion, we transform the density into a RED. The fusion is then conducted on the RED in ellipse parameter space.

To handle estimation, we proposed to replace the squared Euclidean distance with the squared GW distance, i.e., we want to consider the actual shape of the ellipse when finding the best estimate. However, minimizing the GW distance is not trivial (red arrow in Figure 3). To overcome this challenge, we introduced the square root space, consisting of center, the elements of the shape matrix' square root, and kinematics. We transform the density into this space via particle approximation. The property of the square root space we want to utilize is that the expectation, i.e., the average of the particle density, minimizes the squared ESR distance, which is an approximation of the squared GW distance. Therefore, we can get an approximated MMGW estimate easily using the approximation via the square root space (blue arrows in Figure 3). Furthermore, it is of note that if only 1 estimate is present and no fusion is needed, the transformation into square root space can also be used directly from the regular density and not the RED to get the MMGW estimate.

IV. Implementations

With the result of the previous section, we end up with a density on ellipses in ellipse parameter space, the RED,

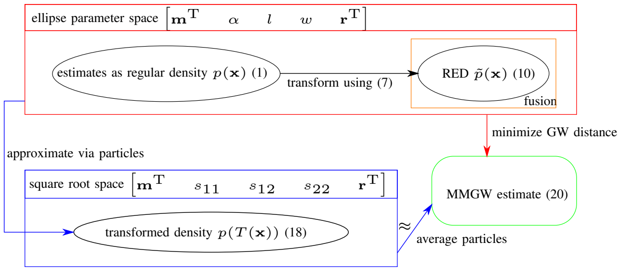

ellipse parameter space $[\mathbf{m}^T \quad \alpha \quad l \quad w \quad \mathbf{r}^T]$
estimates as regular density $p(\mathbf{x})$ (1)
transform using (7)
RED $\tilde{p}(\mathbf{x})$ (10)
fusion
minimize GW distance
approximate via particles
square root space $[\mathbf{m}^T \quad s_{11} \quad s_{12} \quad s_{22} \quad \mathbf{r}^T]$
MMGW estimate (20)
transformed density $p(T(\mathbf{x}))$ (18)
≈
average particles

Fig. 3: A summary of the two spaces and the densities we consider and how the MMGW estimate can be gained from them.

and a transformation of the density to calculate the estimate with respect to a distance measure on ellipses, the squared GW distance. In this section, we present two different fusion approaches based on these concepts. First, we provide an approximation to fuse in ellipse parameter space in Section IV-A using the RED. Second, we briefly recap the fusion method used in our previous work in Section IV-B.

A. Ellipse Parameter Space

For the fusion in ellipse parameter space, each component of the prior RED needs to be multiplied with each component of the likelihood RED

$$\begin{aligned}\tilde{p}(\mathbf{x}|\hat{\mathbf{x}}_s) &\sim \tilde{p}(\hat{\mathbf{x}}_s|\mathbf{x}) \cdot \tilde{p}(\mathbf{x}) \\ &= \sum_{k=-\infty}^{\infty} p_t(K_k(\hat{\mathbf{x}}_s)|\mathbf{x}) \cdot \sum_{j=-\infty}^{\infty} p_t(K_j(\mathbf{x})) \\ &= \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} p_t(K_k(\hat{\mathbf{x}}_s)|\mathbf{x}) \cdot p_t(K_j(\mathbf{x})) \ , \qquad (21)\end{aligned}$$

with the orientations $\alpha_{\mathbf{x}}$ and $\alpha_{\hat{\mathbf{x}}_s}$ restricted as in (10). The sums can be simplified using the 2π periodicity of the orientation, so we can write (21) as

$$\tilde{p}(\mathbf{x}|\hat{\mathbf{x}}_s) \approx c_1 \cdot \sum_{j \in \mathbb{J}} \sum_{k \in \mathbb{K}} p_t(K_k(\hat{\mathbf{x}}_s)|\mathbf{x}) p_t(K_j(\mathbf{x})) \ , \qquad (22)$$

with normalizing constant c_1, $\mathbb{J} = \{n, n+1, n+2, n+3\}$ with $n \in \mathbb{Z}$, so that all four orientations are used, and $\mathbb{K}$ analogous and chosen in a way that for each $i \in \mathbb{J}$, there is a $k \in \mathbb{K}$ which minimizes the difference in their α parameter. The last condition is necessary to avoid the innovation in α being bigger than π or smaller than $-\pi$. We further approximate the components as Gaussians, assuming the probability mass of l and w below 0 to be minor. Thus, we get the prior and likelihood as Gaussians as described in Section III

$$\begin{aligned} p_t(K_k(\hat{\mathbf{x}}_s)|\mathbf{x}) &\approx p(K_k(\hat{\mathbf{x}}_s)|\mathbf{x}) \ , \qquad (23) \\ p_t(K_j(\mathbf{x})) &\approx p(K_j(\mathbf{x})) \ . \qquad (24)\end{aligned}$$

We end up with 16 components, each individual update conducted in a Kalman fashion. With consecutive update steps, the number of components grows, so we apply a mixture reduction. The MMGW estimate is then approximated by sampling particles from the RED, transforming them as in (18), and then averaging them (blue arrows in Figure 3). Pseudocode can be found in Algorithm 1. As we fuse using REDs and determine the mean via MMGW estimation, this method is called Random Ellipse Density Minimum Mean Gaussian Wasserstein (RED-MMGW).

```
input : multimodal global RED with means $\hat{\mathbf{x}}$, covariances $\mathbf{C}$, and
        weights $\mathbf{w}$, local sensor estimate's mean $\hat{\mathbf{x}}_s$ and covariance
        $\mathbf{C}_s$,
output: updated RED means $\hat{\mathbf{x}}^+$, covariances $\mathbf{C}^+$, and weights $\mathbf{w}^+$
for $k \in \{0, 1, 2, 3\}$ do
    $\hat{\mathbf{x}}_s.k \leftarrow$ K($\hat{\mathbf{x}}_s$, $k$)
    $\mathbf{C}_s.k \leftarrow$ K($\mathbf{C}_s$, $k$)
end
for $n \in [0,$len($\mathbf{w}$)$]$ do
    for $k \in \{0, 1, 2, 3\}$ do
        $\hat{\mathbf{x}}^+.(4n+k)$, $\mathbf{C}^+.(4n+k)$, $\mathbf{w}^+.(4n+k)$
          $\leftarrow$kalman($\hat{\mathbf{x}}.n$, $\mathbf{C}.n$, $\hat{\mathbf{x}}_s.k$, $\mathbf{C}_s.k$)
        $\mathbf{w}^+.(4n+k) \leftarrow \mathbf{w}^+.(4n+k) \cdot \mathbf{w}.n$
    end
end
$\hat{\mathbf{x}}^+$, $\mathbf{C}^+$, $\mathbf{w}^+$ $\leftarrow$reduce_mixture($\hat{\mathbf{x}}^+$, $\mathbf{C}^+$, $\mathbf{w}^+$)
return $\hat{\mathbf{x}}^+$, $\mathbf{C}^+$, $\mathbf{w}^+$
```

Algorithm 1: The RED fusion algorithm relies on the following functions: K() switches according to (7) with the last input as k when given state means and switches covariances accordingly as well, len() provides the length of the input vector, kalman() is the Kalman filter correction (restricting the innovation in α between $-\pi$ and π) providing updated mean, covariance, and likelihood, and reduce_mixture() is a mixture reduction function also providing normalized weights.

B. Square Root Space

In our previous work [28], we utilized the transformations properties

$$\begin{aligned} T(K_0(\mathbf{x})) &= T(K_k(\mathbf{x})) \quad && \forall k \in \mathbb{Z} \ , \qquad (25) \\ T(\hat{\mathbf{x}}_1) &= T(\hat{\mathbf{x}}_2) && \text{if } l_1 = w_1 = l_2 = w_2 \\ & && \wedge \, \mathbf{m}_1 = \mathbf{m}_2 \ , \qquad (26)\end{aligned}$$

to transform the density before fusion. This idea uses another property of the square root space, that the ambiguities are not present anymore. Looking at Figure 3, we start off with the estimate and instead of transforming into a RED, we directly go the first blue arrow to transform into square root space. To conduct a fusion in this space, we approximate the transformed density as a Gaussian with mean $\hat{\mathbf{y}}$ and covariance $\mathbf{D}$ and use a Kalman filter for the fusion. The MMGW estimate can then be approximated as the mean of the posterior density. The approximation is conducted by drawing m particles

$$\mathbf{p}^{(j)} \sim \mathcal{N}(\hat{\mathbf{x}}, \mathbf{C}) \quad j \in \{1, ..., m\} \ . \qquad (27)$$

The particles will be in ellipse parameter space (1). We then transform each individual particle and approximate the

```
input : Gaussian approximated global mean $\hat{\mathbf{y}}$ and covariance $\mathbf{D}$ in
        square root space, local sensor estimate's mean $\hat{\mathbf{x}}_s$ and
        covariance $\mathbf{C}_s$ in ellipse parameter space, and the number
        of particles $m$
output: updated mean $\hat{\mathbf{y}}^+$ and covariance $\mathbf{D}^+$
for $j \in [0, m]$ do
    $p.j \leftarrow$ mvn($\hat{\mathbf{x}}_s$, $\mathbf{C}_s$)
end
$\hat{\mathbf{y}}_s \leftarrow$mean(T($p$))
$\mathbf{D}_s \leftarrow$mean(outer(T($p$) $-\hat{\mathbf{y}}_s$, T($p$) $-\hat{\mathbf{y}}_s$))
$\hat{\mathbf{y}}^+$, $\mathbf{D}^+ \leftarrow$kalman($\hat{\mathbf{y}}$, $\mathbf{D}$, $\hat{\mathbf{y}}_s$, $\mathbf{D}_s$)
return $\hat{\mathbf{y}}^+$, $\mathbf{D}^+$
```

Algorithm 2: The MC-MMGW algorithm relies on the following functions: mvn() provides a sample from a multivariate normal distribution, T() describes the transformation from (18), mean() provides a mean, outer() describes the outer product, and kalman() is a regular Kalman update.

transformed particle density as a Gaussian distribution

$$\hat{\mathbf{y}} \approx \frac{1}{m}\sum_{j=1}^{m} T(\mathbf{p}^{(j)}) \ , \tag{28}$$

$$\mathbf{D} \approx \frac{1}{m}\sum_{j=1}^{m} (T(\mathbf{p}^{(j)}) - \hat{\mathbf{y}})(T(\mathbf{p}^{(j)}) - \hat{\mathbf{y}})^{\mathrm{T}} \ . \tag{29}$$

The local estimate $\hat{\mathbf{x}}_s$ with covariance $\mathbf{C}_s$ is transformed analogously. The fusion is then conducted based on the Kalman filter formulas. This method using Monte Carlo approximation of the density is called Monte Carlo Minimum Mean Gaussian Wasserstein (MC-MMGW) with pseudo-code provided in Algorithm 2. It should be noted here that this method is related to fusing the shape matrices. It uses the square roots however (due to the aforementioned properties of the square root space) and transforms not just the estimates, but the entire density, preserving additional information from the covariances. In the next section, we show that the Gaussian approximation of the density is not enough to deal with the non-linearities in certain scenarios and that the fusion via REDs improves the results.

V. Experiments

To support our findings, we provide experiments for a comparison of the MMSE estimate using Euclidean distance, the MMGW estimate including its approximation using ESR distance, and a simple averaging of the shape matrices in Section V-A. Additionally, we evaluate the RED-MMGW and the MC-MMGW in a simulation with a moving ellipse in Section V-B, comparing them to the state-of-the-art. The source code for the experiments is publicly available[1].

A. Ellipse estimation

In this section, we provide a more detailed evaluation of the MMGW estimator, by comparing different approaches for determining point estimates of a Gaussian distributed posterior density on ellipses. The first naive approach is the mean as it minimizes the mean squared Euclidean error on the original ellipse parameter space. Our proposed method, the approximated MMGW estimate from Section III-B, minimizes the squared ESR distance. In essence, it approximates the state density via particles, transforms them into square root matrix space via (18), and averages (blue arrows in Figure 3). To highlight the difference of just averaging the shape matrices instead of their square roots, which might seem more intuitive given a particle density of shape matrices, we include this approach as well under the name Euclidean distance in shape space, in contrast to the Euclidean distance in (ellipse) parameter space. Finally, we provide the MMGW estimate as the optimal solution, minimizing the squared GW distance, which we proposed as a more appropriate metric on ellipses than the squared Euclidean distance. For the calculation, we utilized the optimization described by [56] with initially equal weights and the mean from the ESR distance as initial guess.

	Euclidean params	Euclidean shape	ESR	GW
Test 1	1.5888	1.6043	1.5834	1.5817
Test 2	4.2849	3.7328	3.6477	3.6458
Test 3	4.8260	3.7834	3.6917	3.6916

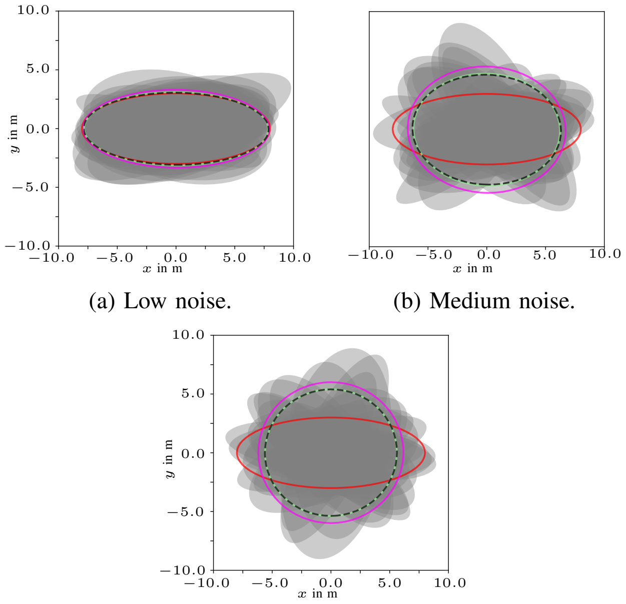
TABLE I: Squared GW error of MMSE estimate using Euclidean distance in parameter space, Euclidean distance in shape matrix space, the approximated MMGW estimate using ESR distance, and the MMGW estimate using GW distance. The first test uses low, the second medium, and the third high orientation noise.

We conducted three experiments. For the first, we took an ellipse estimate with mean $\hat{\mathbf{x}} = \begin{bmatrix}0 & 0 & 0 & 8 & 3\end{bmatrix}^{\mathrm{T}}$ and covariance $\mathbf{C} = diag(\begin{bmatrix}0.5 & 0.5 & 0.01\pi & 0.5 & 0.5\end{bmatrix})$, for the second one, we modified the orientation noise to 0.2π, and for the third to 0.5π, because the differences are most prominent with higher orientation uncertainty. The state consists of the 2D center, orientation, length, and width. As the estimator focuses on the shape, the kinematics are omitted for these tests (see also Section III). In all cases, we drew 1000 particles to approximate the density for the estimators and to approximate the mean squared GW error of each estimate. The results can be found in Table I and Figure 4.

We find that the MMGW estimator approximated using ESR distance is quite accurate with respect to the actual MMGW estimate obtained via optimization. This coincides with the findings from Appendix A. Another important aspect is the relation to the Euclidean distance. For low orientation noise, the estimates are similar, but for high noise, i.e., if we have very low information on the actual orientation, the GW based estimators tend to a more circular form, which looks more intuitive as a high orientation noise means the true ellipse could be near 90 degree to the Euclidean estimate, making a circular estimate more reasonable if the actual area of the ellipse is considered. In other words, the Euclidean estimate we get as the mean of our Gaussian distributed state might seem intuitive at first, but the higher the noise, the further away it is from minimizing the squared GW distance.

Another interesting comparison is the Euclidean mean in shape matrix space, calculated as the mean of the particles' shape matrices. While this seems to be an intuitive estimate regarding a particle density of shape matrices, it can be seen

[1] https://github.com/Fusion-Goettingen/



(a) Low noise.

(b) Medium noise.

(c) High noise.

Fig. 4: MMSE estimates based on Euclidean distance in parameter space (red), Euclidean distance in shape matrix space (magenta), ESR distance (green), and GW distance (dotted black) for different orientation noises. 20 sample particles to highlight uncertainty are drawn in grey.

that it is slightly larger than the MMGW estimate and even worse than the Euclidean mean in low orientation noise. This phenomenon is further elaborated in Section VI-B.

B. Shape Fusion

For the evaluation of the fusion algorithms, the methods described in Section IV are compared with each other and with the state-of-the-art. This includes the regular fusion with Euclidean distances and random matrix based fusion approaches. Regarding the latter, a direct comparison is difficult however, as we assume that sensors provide estimates parameterized as (1), not as random matrices and not as measurement points, i.e., we assume the sensor to be a black box offering us mean and covariance of an ellipse as discussed in Section II-B. To still offer a baseline, we transform the ellipse estimates into a shape matrix and handle all of them as Random Matrices with the same degree. We then utilize the method described by [18], giving all sensor estimates the same weight. Each random matrix gets the degree 6 and to consider a forget model, we utilize the version of the predict function from [57]. We call this method simply "shape mean". More details on the relation to random matrix based distributed fusion approaches can be found in Section VI-B. While the method from [33] also considers orientation and semi-axes, it is not considered here. The reason is that we focus on the shape fusion itself, i.e., we do not consider point clouds to be shared. Excluding the point clouds would reduce the method to the averaging of the estimates' shape matrices, which boils down to the "shape mean" method, but without the forget model. Removing the forget model would be counter productive however, as the shape (specifically the orientation) can change. Therefore, we consider it not appropriate for our setting.

For the MC-MMGW, we used 1000 particles. For the RED-MMGW, we apply mixture reduction, discarding components with low weights, merging those close to each other, and pruning unlikely components to ensure the number of components is below a threshold. On the result, we apply the MMGW estimator (using 1000 particles to approximate the transformed density). We also utilize the MMGW estimator on the posterior of the standard Kalman filter, named here Regular-MMGW. To make the scenario more realistic, we consider moving ellipses, utilizing a Nearly Constant Velocity (NCV) model, i.e., the kinematic part is a Cartesian velocity $\mathbf{r} = \dot{\mathbf{m}}$.

The experiments provide the methods with a prior from which the ground truth is sampled in each run. We simulate sensors with different uncertainties in the semi-axis length and width, i.e., the sensor has a higher uncertainty in the direction it is facing. In addition, the sensors use different orientations for the ellipse to include the problem of ambiguous parameterization (as discussed in Section III). Before calculating an estimate, the simulated sensor switches the parameterization by a random integer k according to (7). The sensors provide ellipse estimates (drawn from the ground truth) and their covariances. We conduct three experiments with *low*, *medium*, and *high* orientation noise. For each experiment, we conducted 1000 Monte Carlo runs and plotted the convergence of the squared GW error over 20 time steps, i.e., 20 local estimates are provided over time. Each time step is 1 second. To also provide a shape change, the ellipse will simply turn along with the velocity. The axes of the ground truth stay constant between time steps. An example run with estimates from the RED-MMGW can be seen in Figure 5, with only every other time step plotted for better visibility.

The prior mean $\hat{\mathbf{x}} = \begin{bmatrix}0 & 0 & 0 & 8 & 3 & 10 & 0\end{bmatrix}^{\mathrm{T}}$ is used and to model that no prior knowledge is available for the orientation, the prior covariance matrix $\mathbf{C} = \mathit{diag}(\begin{bmatrix}0.5 & 0.5 & 0.5\pi & 0.5 & 0.5 & 10.0 & 10.0\end{bmatrix})$ is chosen with a relatively high orientation uncertainty. After the ground truth is drawn from the prior and after each prediction step, the orientation and velocity vector are aligned with each other. We conduct three tests with three sensor noise settings. One is sensor *low* with covariance $\mathbf{C}_s = \mathit{diag}(\begin{bmatrix}0.5 & 0.5 & 0.01\pi & 0.5 & 0.1 & 0.05 & 0.05\end{bmatrix})$ (similar to Figure 4a). Note that before drawing the estimates from the ground truth, the orientation and semi-axes are shifted according to (7) as described above, resulting in the noise on the semi-axes being actually different between the sensors. For sensor *medium* the variance of the orientation in the covariance is set to 0.2π (similar to Figure 4b) and for *high*, it is 0.5π (similar to Figure 4c). As the shape mean does not consider the estimates' covariances and to cope with the change in orientation of the true ellipse during the movement, we modified the forget parameter τ in its prediction step, providing 0.2 in *low* noise, 5.0 in *medium* noise, and 10.0 in *high* noise, which produced the best results in our tests.

The results can be found in Figure 6. Note that the squared GW distance is used as the error function.

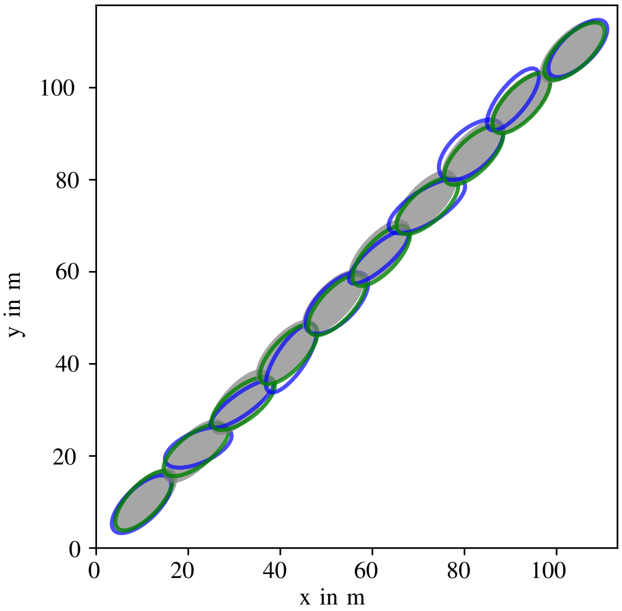


Fig. 5: Sample run under low orientation noise with ground truth in grey, sensor estimates in blue, and the centrally fused RED-MMGW estimate in green for every other time step.

The results of test 1 in Figure 6a show the problems of the regular fusion with the ambiguous parameterization clearly. The shape mean converges, but it does not consider the different uncertainties in the semi-axes and is overall worse than the RED-MMGW and MC-MMGW, which deal with the ambiguity very well.

The results of test 2 in Figure 6b demonstrate the advantages of the RED fusion, dealing with the ambiguities and incorporating information about different axes uncertainties. Especially in this case, it can also be seen how MC-MMGW is outperformed by RED-MMGW, as the approximation of the transformed density as a Gaussian fails under the increased noise.

Finally, the results of test 3 in Figure 6c show an interesting phenomenon. As the MMGW based methods estimate a circular target to deal with the low amount of orientation information, which actually minimizes the squared GW distance, the regular fusion also seemingly improves, as the estimate is also more round. It must be noted here that the reason for that is the mixing of the axes due to ambiguity (see also Figure 2), so the result is good by coincidence. The same holds for the MC-MMGW, as the approximation also leads to a round estimate, explaining the improvements compared to test 2.

Given the Kalman filter-based linear fusion and the shape mean as the state-of-the-art in fusion of elliptical estimates parameterized by orientation and semi-axes, we demonstrated that our method offers improvements of up to 40% with respect to the squared GW distance.

Due to page constraints, we excluded tests with no ambiguous parameterizations. In these cases, where semi-axes and orientation are always correctly assigned and thus can be seen as simple Gaussian distributed random variables, the Kalman filter is of course the best choice. However, using the MMGW estimator on the Kalman filter posterior still provides a better estimate with respect to the squared GW distance, as is demonstrated in Section V-A.

VI. Discussion

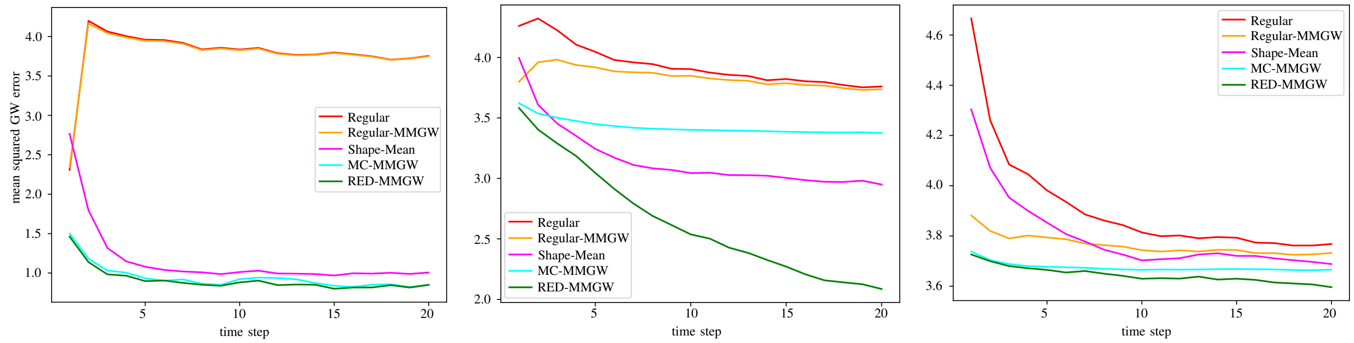
In this section, we provide further insights on different behavior and properties of the discussed methods, comparing the MMGW estimation with the minimization of the Euclidean distance on ellipse parameters in Section VI-A and with the minimization of the Euclidean distance in shape matrix space in Section VI-B.

A. GW vs Euclidean distance on ellipse parameters

In this article, we demonstrated in simulations that in scenarios with high orientation noise, the Euclidean mean provides worse results with respect to the squared GW distance compared to the MMGW estimate (and the regular Kalman filter provides worse results than the Regular-MMGW Kalman filter). We further elaborate this difference by providing the MMSE estimates using Euclidean and ESR distance in low and high orientation noise scenarios. In the case of an uncertain orientation, the methods based on Euclidean distance would keep the semi-axes and average the orientation. With low information on the orientation, this can provide poor results. The ESR based methods systematically deal with this issue by providing more circular shaped estimates. Depending on the scenario, this may be more desirable. Such a situation could happen, e.g., if the semi-axis uncertainty was low (due to long tracking or prior knowledge) and the target would turn under high sensor noise. As the target's dimensions are certain, the uncertainty would be reflected in the orientation. As a result, the estimate could be turned 90 degree in relation to the ground truth. Especially with long ellipses, this would lead to a high squared GW error. The advantage of the GW distance here is that, unlike the Euclidean distance, it does not just consider the individual state parameters, but interprets the state as a shape and considers the area of that shape (this is demonstrated fairly well in Figure 4c). This leads to the more circular estimate, which minimizes the squared GW distance. Note here that the certain estimates of the semi-axes in this example are still preserved in the state density and can be utilized if the orientation uncertainty gets smaller during an update. Only the estimate is circular.

This brings us to an important parallel between the regular estimation and the MMGW based approaches, which occurs when the shape matrices commute, i.e., if the axes of the ellipses align. It can be shown that in these cases, the ESR distance would boil down to the Euclidean distance between state vectors containing only center and the overlapping semi-axes

$$\begin{aligned}&\mathrm{ESR}(\mathbf{m_z},\mathbf{Z},\mathbf{m_x},\mathbf{X})^2\\&=||\begin{bmatrix}(\mathbf{m_z}-\mathbf{m_x})^\mathrm{T} & v_k(l_\mathbf{z},w_\mathbf{z})-l_\mathbf{x} & v_{k+1}(l_\mathbf{z},w_\mathbf{z})-w_\mathbf{x}\end{bmatrix}||_2^2\\&\forall\alpha_\mathbf{z},k \text{ with } (\alpha_\mathbf{z}=\alpha_\mathbf{x}+k\frac{\pi}{2}) \text{ and } k\in\mathbb{Z}\ ,\end{aligned}\tag{30}$$



(a) Test 1 (low α noise).

(b) Test 2 (medium α noise).

(c) Test 3 (high α noise).

Fig. 6: Tests comparing different ellipse fusion approaches under different sensor noise.

with $v_k()$ from (8). Now, if the ellipse orientation has a low uncertainty, so the corresponding value in the covariance tends to 0, most particles would nearly commute. As a result, the MMGW estimate would be almost the same as the regular estimate. Additionally, as the approximation of the GW distance via ESR distance would be exact in this case, both estimates would also be optimal with respect to the squared GW distance. This is further demonstrated by the results of Section V-A.

In this context, we also highlight that the standard Kalman filter would provide better results than the RED if the axes association was known, i.e., if there were no ambiguities. This could happen if the orientation and velocity vector were linked. However, as this assumption might not hold true, e.g., the ellipse is used to model a group target or the sensors decouple shape and kinematics or targets are stationary, the RED is still a useful tool to avoid counter-intuitive fusion results due to wrongly assigned semi-axes.

While the focus of this article is the fusion of estimates, we still want to comment on data association, another important topic of multi-sensor fusion, e.g., in a multi-target context. As it did for the fusion itself, ambiguity can lead to counter-intuitive results for the association as well, leading to the possibility of the distance between the same ellipse being unequal to zero (see Figure 2). The GW distance can solve this, but the entire density needs to be considered. Transforming it as is done in MC-MMGW or using REDs can both tackle this challenge, but an evaluation and comparison for this usage is future work.

B. GW vs Euclidean distance on shape matrix space

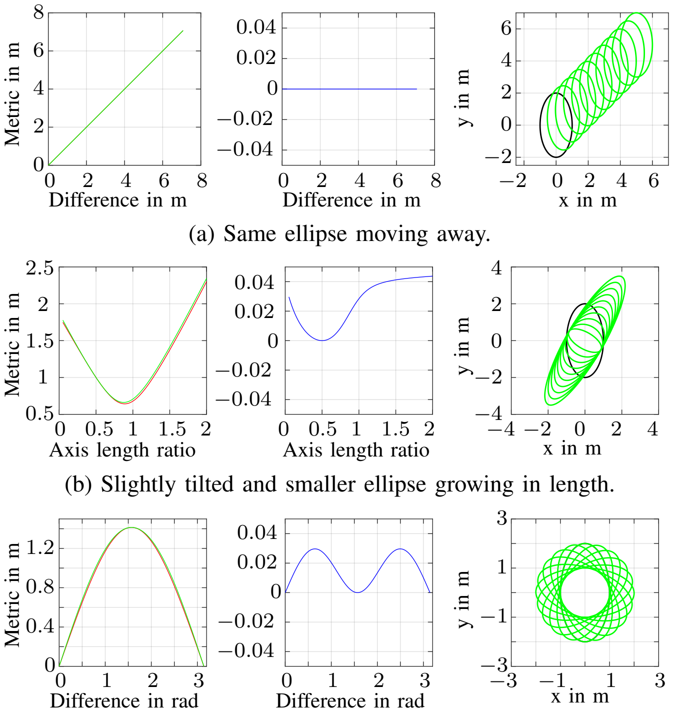
The particles from an empirical density in matrix square root space can simply be averaged to compute the approximated MMGW estimate. Note that [30] calculate the weighted sum of their particle density using the shape matrices, not their square roots, finding the estimate which minimizes the squared Euclidean distance between the matrices instead. Similarly, [33] combine estimates from different sensors by turning them to shape matrices and then calculating their weighted average. However, based on our experience, using the shape matrices' square roots provides better results with respect to the Gaussian Wasserstein distance and the ellipse parameters. This can be demonstrated with a simple example regarding commute matrices. As shown in (30), averaging the square roots in this special case boils down to averaging the lengths and widths. However, averaging the shape matrices would mean averaging the squared lengths and widths. This provides ellipses which are too large, which can be seen in Figure 4. For example, if two equally likely estimates with lengths 2 and 10 are drawn, averaging the square root matrices results in a length estimate $\frac{2+10}{2} = 6$ and averaging the shape matrices produces $\sqrt{\frac{4+100}{2}} \approx 7$.

We also want to discuss the difference to random matrix based fusion approaches. In general, as these approaches employ an Inverse-Wishart distribution to represent an ellipse, their applicability on ellipses parameterized with orientation and semi-axes is limited.

The method from [29] is made for merging of shapes, also increasing their size if both shapes are further apart which is a different type of application. The weighted average to minimize the Kullback-Leibler divergence [18] does not use the estimates' covariances, i.e., it is not suitable if they differ between estimates. The method from [30] utilizes the measurement point clouds directly, which we do not consider is this article. [31] provide a method sharing Gaussian mixture approximations of the density transformed into the same parameterization we use between sensors. However, the Gaussian mixtures are generated by sampling from Random matrices based on an importance distribution and weighting them with their likelihoods, while our approach assumes elliptical estimates and their covariances are given. These estimates cannot be directly translated to those Gaussian mixtures. Hence, a direct comparison with our work is not feasible. The same holds for the latent variable approximations in [32].

VII. Conclusion and Future Work

In this work, we proposed a novel Bayesian framework for estimation and fusion of ellipse densities represented by center, orientation, and semi-axis length and width. The main concepts are the identification of a density on ellipses, the RED, and the utilization of the squared GW distance to create a MMGW estimator. Approximating the GW distance with the ESR distance, we derived an estimator which can be calculated via averaging of particles. In simulations, we demonstrated the



(a) Same ellipse moving away.

(b) Slightly tilted and smaller ellipse growing in length.

(c) Same ellipse rotating.

Fig. 7: Comparison of squared GW and squared ESR distance with metric output on the left (red: GW, green: ESR), metric difference in the middle, and ground truth with every tenth iteration of the estimate on the right (black: ground truth, green: estimates).

robustness of this approximation and compared our methods to state-of-the-art algorithms in respect to estimation and fusion. We highlighted the advantages of our approach, those being the inclusion of the ellipse parameter state's covariance, the dealing with ambiguous parameterizations of ellipses, and more intuitive estimates in scenarios with high orientation noise. In summary, the RED provides a suitable way of handling the ambiguous parameterization of ellipses during fusion if the parameterization of the ellipse is unclear. The MMGW estimator provides an intuitive estimate based on the posterior density, RED or not.

For future work, we intend to further improve the mixture reduction of RED based fusion. Regarding the Monte Carlo-based approach for fusion in square root space, we seek to better preserve the transformed density by means of direct multiplication of particle densities [58]. As for other shapes, we want to investigate appropriate metrics for rectangles and test if the MMGW and RED concepts also work for them due to the equal parameterization. Regarding more complex contours, like star-convex shapes, we wish to determine whether we can apply the same principle of finding an appropriate distance measure and density representation here as well. As data association is also an important topic in multi-sensor fusion, we also wish to investigate the use of GW distance instead of Euclidean distance in this area. An extension to 3D is also an interesting topic. The REDs would face additional challenges, i.e., more possibilities of representing the same ellipsoid and the issue of the corner case in 2D, that a circle has an arbitrary orientation, would be provoked for one of the three angles if only two of the three axes were equal. For the MMGW estimate however, the ESR distance can easily be used in 3D.

Appendix

A. Comparison of GW and ESR distance

For the comparison, we utilized the experiments from [26] with the GW and the ESR distances only (see Figure 7). For the first experiment, the ellipses' orientations are the same. With the approximation of the GW distance by the ESR distance being exact in this case, the two metrics behave the same (see Figure 7a). For the second experiment, the second ellipse is tilted slightly. A difference can thus be seen here, getting larger with the ellipse's length increasing beyond the ground truth's length, but except for that shift, they still behave similar (see Figure 7b). Finally, for the rotation experiment, the general behavior is also the same, with the difference between the metrics being greatest when the angle offset between the ellipses is $\frac{\pi}{4}$ shifted by a multiple of $\frac{\pi}{2}$ (see Figure 7c), so when the angle is in the middle of two orientations which would make the estimate and the ground truth commute.

References

[1] L. Mihaylova, A. Carmi, F. Septier, A. Gning, S. Pang, and S. Godsill, "Overview of Bayesian Sequential Monte Carlo Methods for Group and Extended Object Tracking," *Digital Signal Processing*, vol. 25, pp. 1–16, Feb. 2014.

[2] K. Granström, M. Baum, and S. Reuter, "Extended Object Tracking: Introduction, Overview and Applications," *ISIF Journal of Advances in Information Fusion*, vol. 12, no. 2, Dec. 2017.

[3] 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, Oct. 2005.

[4] K. Granström, C. Lundquist, and U. Orguner, "Tracking Rectangular and Elliptical Extended Targets Using Laser Measurements," in *Proceedings of the 14th International Conference on Information Fusion (Fusion 2011)*, Chicago, Illinois, USA, Jul. 2011.

[5] K. Schueler, T. Weiherer, E. Bouzouraa, and U. Hofmann, "360 Degree Multi Sensor Fusion for Static and Dynamic Obstacles," in *IEEE Intelligent Vehicles Symposium*, 2012, pp. 692–697.

[6] M. Feldmann, D. Fränken, and W. Koch, "Tracking of Extended Objects and Group Targets using Random Matrices," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1409–1420, 2011.

[7] S. Yang and M. Baum, "Tracking the Orientation and Axes Lengths of an Elliptical Extended Object," *IEEE Transactions on Signal Processing*, 2019.

[8] P. Brosseit, B. Duraisamy, and J. Dickmann, "The Volcanormal Density for Radar-Based Extended Target Tracking," in *IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Oct 2017, pp. 1–6.

[9] J. Lan and X. R. Li, "Tracking of Extended Object or Target Group Using Random Matrix: New Model and Approach," *IEEE Transactions on Aerospace and Electrical Systems*, vol. 52, no. 6, pp. 2973–2988, Dec. 2016.

[10] ——, "Tracking of Maneuvering Non-Ellipsoidal Extended Object or Target Group Using Random Matrix," *IEEE Transactions on Signal Processing*, vol. 62, no. 9, pp. 2450–2463, 2014.

[11] M. Baum and U. D. Hanebeck, "Extended Object Tracking with Random Hypersurface Models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, pp. 149–159, Jan. 2014.

[12] N. Wahlström and E. Özkan, "Extended Target Tracking Using Gaussian Processes," *IEEE Transactions on Signal Processing*, vol. 63, no. 16, pp. 4165–4178, 2015.

[13] T. Hirscher, A. Scheel, S. Reuter, and K. Dietmayer, "Multiple Extended Object Tracking Using Gaussian Processes," in *Proceedings of the 19th International Conference on Information Fusion (Fusion 2016)*, July 2016, pp. 868–875.

[14] X. Tang, M. Li, R. Tharmarasa, and T. Kirubarajan, "Seamless Tracking of Apparent Point and Extended Targets Using Gaussian Process PMHT," *IEEE Transactions on Signal Processing*, vol. 67, no. 18, pp. 4825–4838, 2019.

[15] H. Kaulbersch, J. Honer, and M. Baum, "A Cartesian B-Spline Vehicle Model for Extended Object Tracking," in *21st International Conference on Information Fusion (FUSION 2018)*, July 2018.

[16] B. Naujoks, P. Burger, and H.-J. Wuensche, "Fast 3D Extended Target Tracking using NURBS Surfaces," *arXiv preprint arXiv:1909.00767*, 2019.

[17] H. Durrant-Whyte and T. C. Henderson, "Multi Sensor Data Fusion," *Springer handbook of robotics*, pp. 585–610, 2008.

[18] W. Li, Y. Jia, D. Meng, and J. Du, "Distributed Tracking of Extended Targets Using Random Matrices," in *2015 54th IEEE Conference on Decision and Control (CDC)*. IEEE, 2015, pp. 3044–3049.

[19] M. Michaelis, P. Berthold, D. Meissner, and H.-J. Wuensche, "Heterogeneous Multi-Sensor Fusion for Extended Objects in Automotive Scenarios Using Gaussian Processes and a GMPHD-Filter," in *IEEE Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, 2017, pp. 1–6.

[20] K. Granström and U. Orguner, "A PHD Filter for Tracking Multiple Extended Targets Using Random Matrices," *IEEE Transactions on Signal Processing*, vol. 60, no. 11, pp. 5657–5671, Nov. 2012.

[21] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A Multimodal Dataset for Autonomous Driving," *arXiv preprint arXiv:1903.11027*, 2019.

[22] J. S. Fowdur, M. Baum, and F. Heymann, "Tracking Targets with Known Spatial Extent Using Experimental Marine Radar Data," in *2019 22th International Conference on Information Fusion (FUSION)*. IEEE, 2019, pp. 1–8.

[23] 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, July 2011.

[24] W. Koch, "Bayesian Approach to Extended Object and Cluster Tracking using Random Matrices," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 3, pp. 1042–1059, July 2008.

[25] M. Schuster, J. Reuter, and G. Wanielik, "Probabilistic Data Association for Tracking Extended Targets Under Clutter Using Random Matrices," in *2015 18th International Conference on Information Fusion (Fusion)*. IEEE, 2015, pp. 961–968.

[26] S. Yang, M. Baum, and K. Granström, "Metrics for Performance Evaluation of Elliptic Extended Object Tracking Methods," in *Proceedings of the 2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2016)*, Baden-Baden, Germany, Sept. 2016.

[27] I. L. Dryden, A. Koloydenko, and D. Zhou, "Non-Euclidean Statistics for Covariance Matrices, with Applications to Diffusion Tensor Imaging," *The Annals of Applied Statistics*, vol. 3, no. 3, pp. 1102–1123, 2009.

[28] K. Thormann and M. Baum, "Optimal Fusion of Elliptic Extended Target Estimates Based on the Wasserstein Distance," in *22nd International Conference on Information Fusion (FUSION 2019)*, Ottawa, Canada, July 2019. [Online]. Available: https://arxiv.org/abs/1904.00708

[29] K. Granström and U. Orguner, "On Spawning and Combination of Extended/Group Targets Modeled With Random Matrices," *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 678–692, 2013.

[30] G. Vivone, K. Granström, P. Braca, and P. Willett, "Multiple Sensor Measurement Updates for the Extended Target Tracking Random Matrix Model," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 5, pp. 2544–2558, 2017.

[31] J. Liu and G. Guo, "Distributed Asynchronous Extended Target Tracking Using Random Matrix," *IEEE Sensors Journal*, vol. 20, no. 2, pp. 947–956, 2019.

[32] J. Hua and C. Li, "Distributed Variational Bayesian Algorithms for Extended Object Tracking," *arXiv preprint arXiv:1903.00182*, 2019.

[33] Y. Ren and W. Xia, "DISTRIBUTED EXTENDED OBJECT TRACKING BASED ON DIFFUSION STRATEGY," 2020.

[34] J. Hu, L. Xie, and C. Zhang, "Diffusion kalman filtering based on covariance intersection," *IEEE Transactions on Signal Processing*, vol. 60, no. 2, pp. 891–902, 2011.

[35] S. Nilsson and A. Klekamp, "Object Level Fusion of Extended Dynamic Objects," in *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, Sept 2016, pp. 251–258.

[36] M. Herrmann, J. Müller, J. Strohbeck, and M. Buchholz, "Environment Modeling Based on Generic Infrastructure Sensor Interfaces Using a Centralized Labeled-Multi-Bernoulli Filter," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 2414–2420.

[37] B. Duraisamy, M. Gabb, A. V. Nair, T. Schwarz, and T. Yuan, "Track Level Fusion of Extended Objects from Heterogeneous Sensors," in *IEEE Proceedings of the 19th International Conference on Information Fusion (Fusion 2016)*, 2016, pp. 876–885.

[38] M. Michaelis, P. Berthold, T. Luettel, D. Meissner, and H.-J. Wuensche, "A Merging Strategy for Gaussian Process Extended Target Estimates in Multi-Sensor Applications," in *IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 1803–1808.

[39] M. Fröhle, K. Granström, and H. Wymeersch, "Decentralized Poisson Multi-Bernoulli Filtering for Extended Target Tracking," *arXiv preprint arXiv:1901.04518*, 2019.

[40] M. Guerriero, L. Svensson, D. Svensson, and P. Willett, "Shooting Two Birds with Two Bullets: How to Find Minimum Mean OSPA Estimates," *Proceedings of the 13th International Conference on Information Fusion (Fusion 2010)*, 2010.

[41] M. Baum, P. Willett, and U. D. Hanebeck, "On Wasserstein Barycenters and MMOSPA Estimation," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1511–1515, Oct. 2015.

[42] B.-N. Vo, M. Mallick, Y. Bar-Shalom, S. Coraluppi, R. Osborne, R. Mahler, B.-T. Vo, and J. G. Webster, *Multitarget Tracking*. John Wiley & Sons, Inc., 2015.

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

[44] M. Agueh and G. Carlier, "Barycenters in the Wasserstein Space," *SIAM Journal on Mathematical Analysis*, vol. 43, no. 2, pp. 904–924, 2011.

[45] J. Rabin, G. Peyre, J. Delon, and M. Bernot, "Wasserstein Barycenter and its Application to Texture Mixing," in *Scale Space and Variational Methods in Computer Vision*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, vol. 6667, pp. 435–446.

[46] G. Carlier, A. Oberman, and E. Oudet, "Numerical Methods for Matching for Teams and Wasserstein Barycenters," Tech. Rep., May 2014.

[47] M. Cuturi and A. Doucet, "Fast Computation of Wasserstein Barycenters," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, Oct. 2014.

[48] F. Govaers, "On Independent Axes Estimation for Extended Target Tracking," in *2019 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*. IEEE, 2019, pp. 1–6.

[49] H. Zhang, H. Xu, X. Y. Wang, and W. An, "A phd filter for tracking closely spaced objects with elliptic random hypersurface models," in *Proceedings of the 16th International Conference on Information Fusion*, July 2013, pp. 1558–1565.

[50] S. J. Julier and J. K. Uhlmann, "A Non-divergent Estimation Algorithm in the Presence of Unknown Correlations," in *Proceedings of the 1997 American Control Conference (Cat. No. 97CH36041)*, vol. 4. IEEE, 1997, pp. 2369–2373.

[51] L. Chen, P. O. Arambel, and R. K. Mehra, "Estimation Under Unknown Correlation: Covariance Intersection Revisited," *IEEE Transactions on Automatic Control*, vol. 47, no. 11, pp. 1879–1882, 2002.

[52] Y. Bar-Shalom, "On the Track-to-Track Correlation Problem," *IEEE Transactions on Automatic control*, vol. 26, no. 2, pp. 571–572, 1981.

[53] K. V. Mardia and P. E. Jupp, *Directional Statistics*. John Wiley & Sons, 2009, vol. 494.

[54] C. R. Givens and R. M. Shortt, "A Class of Wasserstein Metrics for Probability Distributions," *The Michigan Mathematical Journal*, vol. 31, no. 2, pp. 231–240, 1984.

[55] R. Veltkamp and M. Hagedoorn, "Shape Similarity Measures, Properties and Constructions," in *Advances in Visual Information Systems*, ser. Lecture Notes in Computer Science, R. Laurini, Ed. Springer Berlin / Heidelberg, 2000, vol. 1929, pp. 133–153.

[56] G. Puccetti, L. Rüschendorf, and S. Vanduffel, "On the Computation of Wasserstein Barycenters," *Available at SSRN 3276147*, 2018.

[57] K. Granström, M. Fatemi, and L. Svensson, "Gamma Gaussian inverse-Wishart Poisson multi-Bernoulli Filter for Extended Target Tracking," in *Proceedings of the 19th International Conference on Information Fusion (Fusion 2016)*, Heidelberg, Germany, July 2016.

[58] U. D. Hanebeck, "Bayesian Fusion of Empirical Distributions Based on Local Density Reconstruction," in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2015, pp. 277–282.