Learning-based Extended Object Tracking Using Hierarchical Truncation Measurement Model with Automotive Radar

Xia, Yuxuan; Wang, Pu; Berntorp, Karl; Svensson, Lennart; Granstrom, Karl; Mansour, Hassan; Boufounos, Petros T.; Orlik, Philip V.

TR2021-006 February 09, 2021

Abstract

This paper presents a data-driven measurement model for extended object tracking (EOT) with automotive radar. Specifically, the spatial distribution of automotive radar measurements is modeled as a hierarchical truncated Gaussian (HTG) with structural geometry parameters that can be learned from the training data. The HTG measurement model provides an adequate resemblance to the spatial distribution of real-world automotive radar measurements. Moreover, large-scale radar datasets can be leveraged to learn the geometry-related model parameters and offload the computationally demanding model parameter estimation from the state update step. The learned HTG measurement model is further incorporated into a random matrix based EOT approach with two (multi-sensor) measurement updates: one is based on a factorized Gaussian inverseWishart density representation and the other is based on a RaoBlackwellized particle density representation. The effectiveness of the proposed approaches is verified on both synthetic data and real-world nuScenes dataset over 300 trajectories

IEEE Journal of Selected Topics in Signal Processing

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Learning-based Extended Object Tracking Using Hierarchical Truncation Measurement Model with Automotive Radar

Yuxuan Xia, Pu Wang, Karl Berntorp, Lennart Svensson, Karl Granström, Hassan Mansour, Petros Boufounos, and Philip V. Orlik

***Abstract*—This paper presents a data-driven measurement model for extended object tracking (EOT) with automotive radar. Specifically, the spatial distribution of automotive radar measurements is modeled as a hierarchical truncated Gaussian (HTG) with structural geometry parameters that can be learned from the training data. The HTG measurement model provides an adequate resemblance to the spatial distribution of real-world automotive radar measurements. Moreover, large-scale radar datasets can be leveraged to learn the geometry-related model parameters and offload the computationally demanding model parameter estimation from the state update step. The learned HTG measurement model is further incorporated into a random matrix based EOT approach with two (multi-sensor) measurement updates: one is based on a factorized Gaussian inverse-Wishart density representation and the other is based on a Rao-Blackwellized particle density representation. The effectiveness of the proposed approaches is verified on both synthetic data and real-world nuScenes dataset over 300 trajectories.**

***Index Terms*—Automotive radar, extended object tracking, random matrix, autonomous driving, nuScenes.**

I. Introduction

Automotive radar plays an important role in autonomous driving as it provides reliable environmental perception in all-weather conditions with affordable cost [1]. With the advances in radar technology, modern radar sensors can resolve multiple detection points per object. The tracking of an object that gives rise to multiple detection points is called extended object tracking (EOT). Compared to conventional point object tracking, EOT can lead to improved tracking capability as multiple detections allow us to not only estimate the object kinematic state more precisely but also to infer the object extent state. An overview of EOT literature is given in [2].

To capture the spatial characteristics of automotive radar measurements, it is often assumed that they are spatially distributed as a function of individual measurement likelihoods, also referred to as the *spatial distribution*. Spatial distributions that are discussed in the literature can be divided into two categories: 1) contour models, which reflect the measurement distribution along the object contour, and 2) surface models, which assume that the measurements are generated from the inner surface of an object. For the contour models, typical examples include rectangular shape models [3], [4] and the star-convex shape models [5]–[8]. A widely used surface model is the random matrix approach [9]–[14], which assumes an elliptic object shape. The surface model generally leads to computationally simpler algorithms than the contour model, which enjoys more flexibility to describe more complex shapes.

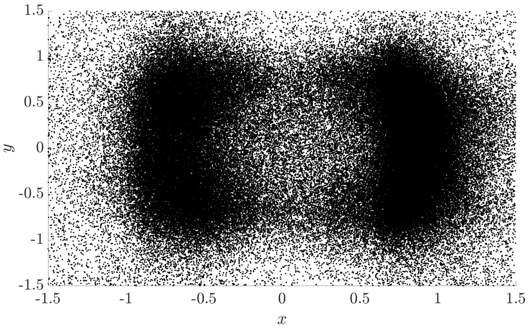


Fig. 1. Accumulated automotive radar measurements of vehicles in a unit frame extracted from the nuScenes dataset [20].

The spatial characteristics of real-world automotive radar measurements are, however, more complex and can neither be well described by the contour model nor by the surface model, see, e.g., [15], [16]. Fig. 1 illustrates how the accumulated radar measurement density is much lower at the center than it is in a vicinity around the outer edges. Measurements also exhibit self-occlusion features: the measurement density is dominant at object parts that are in sight of the sensors. These features of real-world automotive radar measurements have motivated developments of EOT algorithms using *customized* spatial representations with automotive radar. Early efforts include the Set of Points on a Rigid Body (SPRB) models [17], [18] and the direct scattering model [19]. The SPRB models usually require explicit associations between the reflection points and the measurements, whereas the direct scattering model is physics-based and requires certain expert knowledge and manual adaption.

A third category of models, the surface-volume models, have started to attract more attention as they balance between the contour models and the surface models with more realistic features to account for the spatial characteristics of real-world automotive radar measurements. Typical examples of surface-volume models in the literature include the volcanormal measurement model [21], the variational Gaussian mixture model (GMM) [22], [23] and the hierarchical truncated Gaussian (HTG) model [24], [25]. The volcanormal measurement model uses a special donut-shaped spatial density to better approximate the radar measurement spread of vehicles, and the object state is determined using a maximum likelihood (ML)

The work of Y. Xia was partly done during his internship at MERL.

YX, LS and KG are with the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, 41296 Sweden

PW, KB, HM, PB and PO are with Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, 02139 USA.

Corresponding author: pwang@merl.com

estimator. The variational GMM is data-driven and trained using radar datasets. EOT using the learned variational GMMs has been implemented by a Rao-Blackwellized particle (RBP) filter [22] and an extended Kalman filter (EKF) [23]. For the RBP implementation, the kinematic state is represented by particles while the extent state is represented by discrete distributions. For the EKF implementation, the associations between the measurements and the individual Gaussian components is handled using Expectation-Maximization (EM).

In our previous work [24], [25], we proposed an HTG model and integrated it into a random matrix based approach with a modified measurement update step and online truncation bounds estimation for both full-view measurements [24] and partial-view measurements due to self-occlusion [25]. The new random matrix approach for the HTG model leverages the computational advantages of surface models and the customized spatial characteristics of automotive radar measurements. In this paper, we expand on our previous work to offload the ML truncation bounds estimation in the state update by leveraging large-scale public automotive radar datasets. This makes our learning-based method more suitable for EOT with very few measurements, compared to methods using ML estimation of object states [21] or model parameters [4], [24], [25]. Moreover, by having fewer parameters in the learned spatial model, our proposed method can be more robust to measurement model mismatch between the learned measurement model and the true sensor measurements, compared with learning-based methods using variational GMMs.

This paper is based on our previous conference publications [24], [25]. Some preliminary work has also been done in [26], but only with limited derivations and experiments. This paper supplements [24], [25] with more detailed derivations and its new contributions and results are summarized as follows:

1) We present an offline learning method for the geometry-related HTG model parameters using ML estimation and coordinate descent optimization.
2) We integrate the learned HTG model into a random matrix approach based on a factorized Gaussian inverse-Wishart (GIW) density representation.
3) We extend the proposed single sensor (SS) measurement update to multi-sensor (MS) measurement update.
4) We present a modified random matrix based approach of [27] for the HTG model based on a RBP density representation, which offers a trade-off between computational complexity and accuracy.
5) We compare the performance of our proposed methods using the learned HTG model with our previous work [25] in a comprehensive simulation study.
6) We benchmark our proposed methods against two other EOT algorithms [8], [23], using real-world automotive radar measurements from the public nuScenes dataset.
7) The experimental results averaged over 300 trajectories show that our proposed method based on the GIW density has appealing performance in terms of estimation error, computational complexity, and robustness.

The remainder of this paper is organized as follows. The problem formulation is described in Section II. Section III introduces the HTG measurement model and the offline learning of an HTG from an automotive radar dataset. Section IV presents the random matrix approach based on a factorized GIW density for the HTG model for both SS and MS measurement updates. Section V presents a RBP filter implementation for HTG model. Simulation and experimental results are given in Section VI, followed by conclusions in Section VII.

II. Problem Formulation

We consider the problem of single extended object tracking without clutter. The extended object state at time step k is represented as a tuple $\xi_k = (x_k, X_k)$ consisting of a kinematic vector x_k and a symmetric positive definite extent matrix X_k. The kinematic vector x_k encapsulates the center position of the object $[p_{k,x}, p_{k,y}]^{\mathrm{T}}$, object heading ϕ_k, and any other motion parameters of interest, whereas the extent matrix X_k describes the object spatial extension with elliptic form. The square of the lengths of the half axes of the ellipse represented by X_k can be obtained via the eigen-decomposition of X_k.

Radar sensors generally give rise to range/bearing measurements. In this paper, we assume that all the measurements acquired from the automotive radar sensors have already been converted into a two-dimensional Cartesian frame by applying the standard frame conversion [28]. The set of measurements from the sth sensor at time step k is denoted by $Z_k^s = \{z_k^{s,j}\}_{j=1}^{n_k^s}$ where $z_k^{s,j}$ are the independent measurement vectors and n_k^s is the number of measurements for the sth sensor. The total measurements Z_k from the S sensors at time step k is $Z_k = (Z_k^1, \dots, Z_k^s, \dots, Z_k^S)$, the total number of measurements received at time step k is $n_k = \sum_{s=1}^{S} n_k^s$, and the measurement sets up to and including time step k is $Z_{1:k}$.

The objective of object tracking is to recursively compute the posterior density of the object state $p(\xi_k|Z_{1:k})$ using Bayesian estimation. The estimate of the object state ξ_k with corresponding uncertainty measures can then be extracted from the posterior density $p(\xi_k|Z_{1:k})$. Given the posterior density $p(\xi_{k-1}|Z_{1:k-1})$ and the object state transition density $p(\xi_k|\xi_{k-1})$, the predicted density $p(\xi_k|Z_{1:k-1})$ is obtained via the Chapman-Kolmogorov equation

$$p\left(\xi_k|Z_{1:k-1}\right) = \int p\left(\xi_{k-1}|Z_{1:k-1}\right) p\left(\xi_k|\xi_{k-1}\right) d\xi_{k-1}. \quad (1)$$

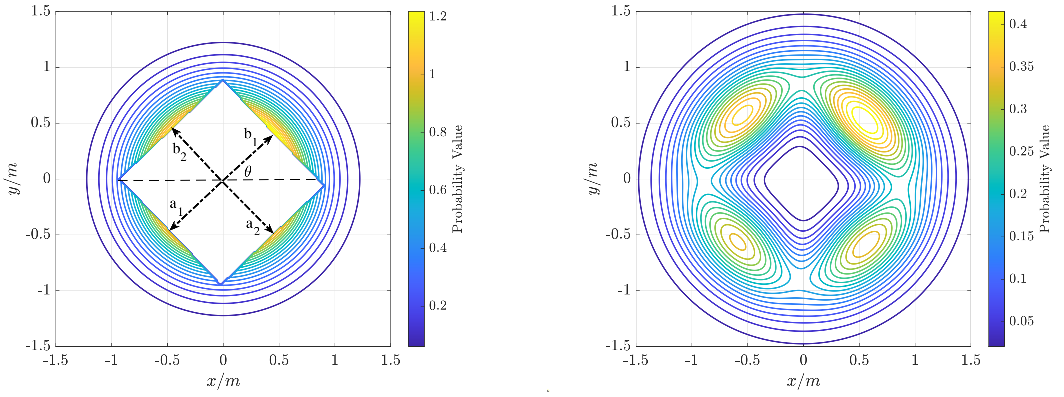
The posterior density $p\left(\xi_k|Z_{1:k}\right)$ is obtained by updating the predicted density $p\left(\xi_k|Z_{1:k-1}\right)$ using measurements Z_k via the Bayes' rule

$$p\left(\xi_k|Z_{1:k}\right) \propto p\left(\xi_k|Z_{1:k-1}\right) p\left(Z_k|\xi_k\right), \quad (2)$$

where $p(Z_k|\xi_k) = \prod_{j=1}^{n_k} p(z_k^j|\xi_k)$ is the joint measurement likelihood with $p(z_k^j|\xi_k)$ denoting the spatial distribution of the measurements. To allow for a recursive use of the prediction and update functions, we seek to approximate the predicted and posterior object state densities such that they all have the same functional form as the initial prior.

III. Hierarchical Truncation Model

In this section, we present the HTG model for modeling the spatial distribution of automotive radar measurements and the HTG model learning from offline training data.



(a)

(b)

Fig. 2. Probability density functions of (a) a truncated Gaussian distribution with parameters $\rho = 0.184$, $\theta = 0.764$ and $B = (0.673, 0.614, 0.670, 0.648)$; and (b) a hierarchical truncated Gaussian distribution obtained by performing (7) on the truncated Gaussian distribution in (a) and a zero-mean Gaussian with covariance $R = M(\theta)\,\mathrm{diag}([0.038, 0.035])M(\theta)^{\mathrm{T}}$.

A. Hierarchical Truncated Gaussian Measurement Model

To capture the spatial characteristics of automotive radar measurements, we propose a hierarchical truncation model for the spatial distribution of the measurements. In our previous work [24], [25], the HTG model is introduced in an *object centered frame*; see Appendix A for illustrative examples. To enable a smooth integration of the HTG model into a learning-based framework, the HTG model here is defined and learned in a *unit frame* that is independent of the object size.

For rigid objects, it is often assumed that the object orientation is aligned with the object heading, such that the object extent matrix X_k can be factorized as

$$X_k = M(\phi_k)E(X_k)E(X_k)^{\mathrm{T}}M(\phi_k)^{\mathrm{T}}, \tag{3}$$

where $E(X_k)$ produces a diagonal matrix of the square root of the eigenvalues of X_k (in decreasing order) and $M(\cdot)$ is the counterclockwise rotation matrix

$$M(\phi_k) = \begin{bmatrix} \cos(\phi_k) & -\sin(\phi_k) \\ \sin(\phi_k) & \cos(\phi_k) \end{bmatrix}. \tag{4}$$

Then, given an object state ξ_k, a measurement z_k defined in the global frame can be expressed using its counterpart u_k defined in a normalized object frame via

$$z_k = Hx_k + M(\phi_k)E(X_k)u_k, \tag{5}$$

where H is the observation matrix that selects the position components $[p_{k,x}, p_{k,y}]^{\mathrm{T}}$ in the kinematic state vector x_k.

We assume that each measurement u_k originates from a measurement source y_k, which is corrupted by a zero-mean Gaussian measurement noise with covariance R_k. We further assume that the measurement source y_k is truncated Gaussian distributed with probability density function (pdf)

$$\mathcal{TN}\left(y_k; 0_{2\times1}, \rho\mathrm{I}_2, D_k\right) = \frac{\mathbf{1}_{D_k}(y_k)}{c_{D_k}}\mathcal{N}\left(y_k; 0_{2\times1}, \rho\mathrm{I}_2\right), \tag{6}$$

where ρ is a scaling factor, $D_k \subseteq \mathbb{R}^2 - \mathrm{rect}(B_k, \theta_k)$ specifies the truncated Gaussian density support, $\mathbf{1}_{D_k}(\cdot)$ is the indicator function on D_k, and c_{D_k} is the normalization factor such that (6) integrates to one. The truncation area $\mathrm{rect}(B_k, \theta_k)$, assumed rectangular, is fully specified by four truncation bounds $B_k = (a_{k,1}, a_{k,2}, b_{k,1}, b_{k,2})$ and an orientation θ_k with respect to the horizontal axis. An illustrative example of $p(y_k)$ is given in Fig. 2 (a). The spatial distribution of u_k is an HTG by marginalizing out the measurement source y_k,

$$\begin{aligned} p(u_k) &= \int p(u_k|y_k)p(y_k)dy_k \\ &= \frac{1}{c_{D_k}}\int_{D_k}\mathcal{N}(u_k; y_k, R_k)\mathcal{N}(y_k; 0_{2\times1}, \rho\mathrm{I}_2)dy_k. \end{aligned} \tag{7}$$

An illustration of $p(u_k)$ is given in Fig. 2 (b). Note that the density support of an HTG distribution is $\mathbb{R}^2$, despite the fact that the density inside the truncation area could be very low. The pseudo code for sampling an HTG density, as illustrated in Fig. 2 (b), is given in Appendix B. To model partial-view measurements caused by self-occlusion, one can set one or more of the truncation bounds to infinity. Also, note that in contrast to the HTG model introduced in [24], [25], the orientation of the truncation area considered in this work does not need to be aligned with the object heading nor the object orientation, i.e., θ can be non-zero. This leads to a better modeling of the spatial characteristics of automotive radar measurements when the vehicle wheels and wheel houses are typical measurement sources; an example is later given in Section VI, Fig. 5(c).

Because the measurements' spatial density differs when the object is observed from different perspectives due to self-occlusion, we may need different HTG models to describe the spatial distributions of different sets of measurements acquired from different sensors. The measurement likelihood, conditioned on the object state ξ_k, the number of measurements n^k and the sensor state, is modeled as[1]

$$p(Z_k|\xi_k) = \prod_{s=1}^{S}\frac{1}{c_{D_k^s}^{n_k^s}}\prod_{j=1}^{n_k^s}\int_{D_k^s}\mathcal{N}(y_k; 0_{2\times1}, \rho^s\mathrm{I}_2) \tag{8}$$

[1] The dependences on the number of measurements and the sensor state are not explicitly expressed here for simplicity. Also, as this paper only considers the problem of single object tracking, modeling the cardinality distribution and estimating the number of measurements is outside the scope of this paper.

$$\times \mathcal{N}\left(E(X_k)^{-1}M(\phi_k)^{-1}\left(z_k^{s,j} - Hx_k\right); y_k, R_k^s\right) dy_k,$$

where D_k^s, specified B_k^s and θ_k^s, is the truncated Gaussian density support for the sth sensor. The parameters of the HTG distribution for the sth sensor include B_k^s, θ_k^s, R_k^s and ρ_k^s.

B. Hierarchical Truncated Gaussian Model Learning

Let us first focus on the problem of learning an HTG model defined in the unit frame. Suppose that the training data to learn the model consist of a set of N two-dimensional accumulated data points $U = \{u^j\}_{j=1}^N$ that have already been transformed to the unit frame, and that we seek to estimate the parameters given the available data. The ML estimate can be obtained by maximizing the joint measurement likelihood with respect to the parameters as

$$\arg\max_{\rho,B,\theta,R} \frac{1}{c_D^N} \prod_{j=1}^{N} \int_D \mathcal{N}\left(u^j; y, R\right) \mathcal{N}\left(y; 0_{2\times 1}, \rho \mathrm{I}_2\right) dy. \quad (9)$$

Computing (9) involves evaluating the convolution of a bivariate truncated Gaussian (cf. (6)) and a bivariate Gaussian, which is intractable in general. To obtain a tractable solution of (9), we assume that the noise covariance R has the same orientation as the truncation area such that

$$R = M(\theta)\, \mathrm{diag}\left([r_1, r_2]\right) M(\theta)^{\mathrm{T}}. \quad (10)$$

This can be regarded as a reasonable assumption because the measurement noise is typically smaller than the object extent. More importantly, such factorization of the measurement noise covariance (10) allows the decomposition of (9) into more explicit expressions to which an optimization solver can be easily applied, as will be explained later. After plugging (10) into (9) and taking the negative log, one can then show that solving (9) is equivalent to obtaining

$$\arg\min_{\rho,B,\theta,r_1,r_2} -\sum_{j=1}^{N} \log \int_D \mathcal{N}\left(M(-\theta)u^j; y, \mathrm{diag}\left([r_1, r_2]\right)\right) \times \mathcal{N}\left(y; 0_{2\times 1}, \rho \mathrm{I}_2\right) dy + N \log c_D \quad (11)$$

where we transform the measurements U to a new frame, such that the orientation of the truncation area is now aligned with the axes and that the noise covariance R after transformation becomes a diagonal matrix. As mentioned previously, a non-zero orientation of the truncation area θ can be utilized to model strong reflection points at object corners. Letting the orientation of the truncation area be zero in the new frame is merely to make the mathematical derivations easier.

With the new parameterization, the integral over the product of two bivariate Gaussians in (11) can be computed as a product of two integrals over univariate Gaussians, and, hence decoupled. With simplifications, we can rewrite (11) as

$$\arg\min_{\rho,B,\theta,r_1,r_2} -\sum_{j=1}^{N} \log \Big(\mathcal{N}\left(\tilde{u}_x^j; 0, r_1+\rho\right) \mathcal{N}\left(\tilde{u}_y^j; 0, r_2+\rho\right) \times \left(1 - \gamma\left(\tilde{u}_x^j, a_1, b_1, r_1\right) \gamma\left(\tilde{u}_y^j, a_2, b_2, r_2\right)\right) \Big) + N \log c_D$$
$$\text{s.t. } \theta \in [-\pi, \pi], \{\rho, B, r_1, r_2\} \in [0, \infty) \quad (12)$$

where

$$\left[\tilde{u}_x^j, \tilde{u}_y^j\right]^{\mathrm{T}} = M(-\theta)u^j, \quad (13a)$$

$$c_D = 1 - \left(\left(\Phi(b_1\rho^{-1/2}) - \Phi(-a_1\rho^{-1/2})\right) \times \left(\Phi(b_2\rho^{-1/2}) - \Phi(-a_2\rho^{-1/2})\right)\right), \quad (13b)$$

$$\gamma(\tilde{u}, a, b, r) = \Phi\left(\frac{b - \frac{\rho\tilde{u}}{r+\rho}}{\sqrt{\frac{r\rho}{(r+\rho)}}}\right) - \Phi\left(\frac{-a - \frac{\rho\tilde{u}}{r+\rho}}{\sqrt{\frac{r\rho}{(r+\rho)}}}\right), \quad (13c)$$

$\Phi(\cdot)$ denotes the cumulative density function of a standard normal distribution and $\{\rho, B, r_1, r_2\} \in [0, \infty)$ is an abuse of notation meaning that all the HTG parameters except θ are non-negative real numbers. The ML estimate of all unknown parameters can be approximately found by applying coordinate descent optimization. The derivation of the cost function in (12) is given in Appendix C, while its optimization over the structural model parameters and numerical performance of the ML estimation are given in Appendix D.

C. Measurement Model Learning for Vehicles

To construct the training data needed in (11), we collect data for which we assume that the object states, the sensor states and the sensor measurements are all available, and that the correspondence between object states and measurements are known. All the sensor measurements can then be transformed to a normalized object frame that is independent of the object states, e.g., positions and sizes.

To account for the self-occlusion feature of automotive radar measurements, we use the aspect angle under which the sensor sees the vehicle to encapsulate the relation between the object state and the sensor state as a single quantity [22]. The aspect angle can be approximately computed as

$$\chi = \phi_{\mathrm{SC}} - \mathrm{atan2}(p_{y,\mathrm{SC}}, p_{x,\mathrm{SC}}) \quad (14)$$

where ϕ_{SC} and $[p_{y,\mathrm{SC}}, p_{x,\mathrm{SC}}]^{\mathrm{T}}$, respectively, denote the object orientation and center position in the sensor frame. The design choice (14) is also used in [22], [23], and an intuitive explanation is that the relative location of the measurements conditioned on similar aspect angles are similar irrespective of the vehicle size or its position in the sensor's field of view.

The training data is formed by accumulating the normalized measurements in the unit coordinate and corresponding aspect angles. The training data is further split into different groups based on the range of the aspect angle, and a conditional HTG model on the aspect angle is learned for each group of data.

IV. Random Matrix Approach for Hierarchical Truncated Gaussian Model

In the random matrix model, the object shape is approximated by an ellipse and the spatial spread of the measurements is modeled by a Gaussian distribution whose covariance is proportional to the object extent matrix. The random matrix appoach was orginally proposed in [9] for noise-free measurements and linear kinematic state based on a coupled GIW density representation. An improved noise modeling was later

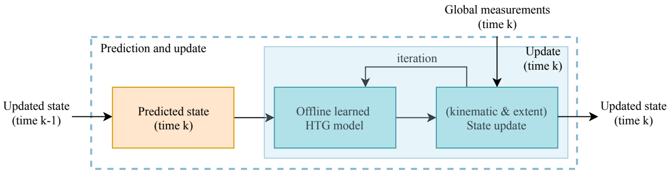


Fig. 3. Diagram of the iterative update method for learned HTG model.

proposed in [10] based on a factorized GIW density representation, which makes the modeling of nonlinear dynamics possible.

In this section, we extend the random matrix approach [10] using a factorized GIW density representation to HTG spatial density. We note that the learned HTG model (8) is defined in a normalized object frame. Therefore, before performing the measurement update, we first need to select a learned HTG model according to the aspect angle and transform it to the global frame using object state ξ_k. Since the object state is unknown, a simple approximate solution is to replace ξ_k with the predicted state estimate $\hat{\xi}_{k|k-1}$, but this may not work well when the prediction is inaccurate. To address this problem, we propose an iterative update method for both SS and MS measurement updates[2].

The complete state update step for the learned hierarchical truncation model runs iteratively over two building blocks: 1) HTG model update, and 2) object state update. See Fig. 3 for a diagram. Specifically, at the tth iteration, we first compute the aspect angle (14) using the updated object state $\hat{\xi}^{t-1}_{k|k}$ at the $(t-1)$th iteration, and then we select a learned HTG measurement model conditioned on $\hat{\xi}^{t-1}_{k|k}$ and convert it to the global frame. From one iteration to the next, a refined HTG model defined in the global frame can be obtained by having a more accurate object state estimate. At the first iteration $(t = 1)$, we may use the predicted state estimate $\hat{\xi}_{k|k-1}$ to initialize the update step by setting $\hat{\xi}^{0}_{k|k} = \hat{\xi}_{k|k-1}$. The iterative update can be run either for a fixed number of iterations, or until some convergence criterion is met. A convergence analysis is carried out in a simulation study in Section VI.

We remark here that, for factorized GIW density, the iterative update was first presented in [11] based on variational Bayesian approximation. In the variational measurement update, the unknown measurement sources are estimated as hidden variables, and one can obtain improved estimates of the measurement sources and object state from one iteration to the next. It is also possible to consider the variational measurement update for HTG spatial density, but this requires further investigation.

For the factorized GIW parameterization, the predicted and posterior densities are, respectively, approximated as

$$\begin{aligned} p(\xi_k|Z_{1:k-1}) &\approx p(x_k|Z_{1:k-1})p(X_k|Z_{1:k-1}) && \text{(15a)}\\ &= \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1})\mathcal{IW}(X_k; v_{k|k-1}, V_{k|k-1}),\\ p(\xi_k|Z_{1:k}) &\approx p(x_k|Z_{1:k})p(X_k|Z_{1:k})\\ &= \mathcal{N}(x_k; m_{k|k}, P_{k|k})\mathcal{IW}(X_k; v_{k|k}, V_{k|k}) && \text{(15b)} \end{aligned}$$

where $\mathcal{IW}(X; v, V)$ denotes an inverse-Wishart (IW) pdf over the matrix X with degrees of freedom v and scale matrix V [30, Def. 3.4.1]. For the pdf $\mathcal{IW}(X; v, V)$ with 2×2 matrices X and V, the expected extent is $\hat{X} = V/(v-6)$. In the literature, several approximate update methods exist for both SS and MS measurement updates [27], [31]. Our proposed update methods for the HTG spatial density are adapted from the SS extent update presented in [10] based on matrix square root computation (FFK)[3] and its MS extension (MSFFK) [27].

A. Single Sensor Update

For the SS $(S = 1)$ case, the random matrix model assumes Gaussian spatial density with measurement likelihood [10]

$$p^{\mathrm{RM}}\left(Z^1_k|\xi_k\right) = \prod_{j=1}^{n^1_k} \mathcal{N}\left(z^{1,j}_k; Hx_k, \rho^1 X_k + R^{1,g}_k\right) \tag{16}$$

where $R^{1,g}_k$ is the measurement noise covariance defined in the global frame. Given that the predicted kinematic state density is also Gaussian, the updated estimate of the centroid kinematics can be determined using the centroid measurement

$$\bar{z}^1_k = \frac{1}{n^1_k}\sum_{j=1}^{n^1_k} z^{1,j}_k, \tag{17}$$

by applying standard (nonlinear) Kalman filter update equations with suitable substitutions of terms [10]. The extent matrix can be approximately updated using two matrices proportional to the measurement spread

$$\bar{Z}^1_k = \sum_{j=1}^{n^1_k}\left(z^{1,j}_k - \bar{z}^1_k\right)\left(z^{1,j}_k - \bar{z}^1_k\right)^{\mathrm{T}}, \tag{18}$$

and the spread of the centroid measurement around the predicted object position, respectively [10].

We note that the HTG spatial density (8) reduces to (16) when $S = 1$ and the integral in (8) is taken over $\mathbb{R}^2$ instead of D^1_k, i.e., when all the truncation bounds are zero. For the HTG spatial density (8), a direct state update using the centroid measurement (17) and the measurement spread (18) is likely to yield inaccurate state estimates due to measurement model mismatch. To improve the estimation performance, we introduce a set of pseudo measurements, such that the spatial density of the union of the received measurements Z^1_k and the pseudo measurements may be regarded as Gaussian. Using the centroid measurement and the measurement spread of the new set of measurements to perform the state update would then yield more accurate state estimates, and at the same time it allows us to make the most use of the available random matrix update equations.

Given truncated Gaussian distributed measurement sources $Y^1_k = \{y^{1,j}_k\}^{n^1_k}_{j=1}$ in a normalized object frame with pdf (6), we observe that if we draw a certain number of samples from the truncated Gaussian distribution whose support is the complement of the support $D^{1,c}_k = \mathbb{R}^2 \setminus D^1_k$, i.e., $y^{1,j,c}_k \sim \mathcal{TN}\left(0_{2\times 1}, \rho^1 \mathrm{I}_2, D^{1,c}_k\right)$, then the drawn samples together with

[2] When multiple sensors are available, it is also possible to use a track-to-track fusion strategy [29], but this is not considered here.

[3] It is named after the authors' initials.

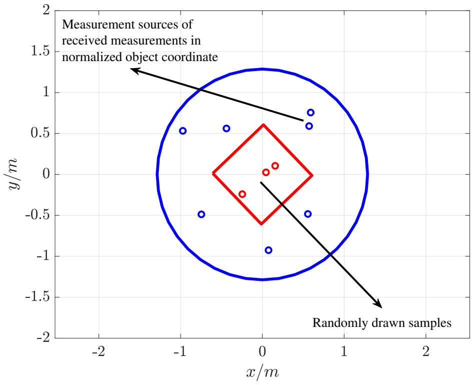


Fig. 4. An example of the measurement sources and drawn samples.

the measurement sources Y_k^1 may be regarded as random samples drawn from the Gaussian distribution $\mathcal{N}\left(0_{2\times1}, \rho^1 \mathrm{I}_2\right)$; see Fig. 4 for an illustration of the pseudo measurements denoted in red circles. In order for this to be justifiable, the relation between the number of measurement sources n_k^1 and the number of samples $n_k^{1,c}$ should satisfy

$$\frac{n_k^1}{n_k^{1,c}} = \frac{c_{D_k^1}}{1 - c_{D_k^1}} \tag{19}$$

where the normalization factor $c_{D_k^1}$ can be interpreted as the probability that a random sample of $\mathcal{N}\left(0_{2\times1}, \rho^1 \mathrm{I}_2\right)$ falls inside the truncation area specified by D_k^1. For each sample $y_k^{1,j,c}$, we further obtain its Gaussian noise corrupted counterpart as $u_k^{1,j,c} \sim \mathcal{N}\left(y_k^{1,j,c}, R_k^1\right)$. Finally, the set of pseudo measurements in the global frame is $Z_k^{1,c} = \{z_k^{1,j,c}\}_{j=1}^{n_k^{1,c}}$ with

$$z_k^{1,j,c} = M(\phi_k)E(X_k)\left(u_k^{1,j,c} + Hx_k\right). \tag{20}$$

Augmented with the introduced pseudo measurements, the set of measurements $Z_k^{1,u} = Z_k^1 \cup Z_k^{1,c}$ can then be regarded as realizations of the spatial density

$$p\left(Z_k^{1,u} | \xi_k\right) = \prod_{z_k^{1,u} \in Z_k^{1,u}} \mathcal{N}\left(z_k^{1,u}; Hx_k, \rho^1 X_k + R_k^{1,g}\right), \tag{21a}$$

$$R_k^{1,g} = M(\phi_k)E(X_k)R_k^1 E(X_k)^{\mathrm{T}} M(\phi_k)^{\mathrm{T}}, \tag{21b}$$

which is in line with the Gaussian spatial density assumption (cf. (16)) used in the random matrix model.

The centroid measurement and measurement spread of $Z_k^{1,u}$ can be, respectively, expressed as

$$\begin{aligned} \bar{z}_k^{1,u} &= \frac{1}{n_k^1 + n_k^{1,c}} \left(\sum_{j=1}^{n_k^1} z_k^{1,j} + \sum_{j=1}^{n_k^{1,c}} z_k^{1,j,c} \right) \\ &= \frac{c_{D_k^1}}{n_k^1} \left(\sum_{j=1}^{n_k^1} z_k^{1,j} + \sum_{j=1}^{n_k^{1,c}} z_k^{1,j,c} \right), \end{aligned} \tag{22a}$$

$$\begin{aligned} \bar{Z}_k^{1,u} &= \sum_{j=1}^{n_k^1} \left(z_k^{1,j} - \bar{z}_k^{1,u}\right)\left(z_k^{1,j} - \bar{z}_k^{1,u}\right)^{\mathrm{T}} \\ &\quad + \sum_{j=1}^{n_k^{1,c}} \left(z_k^{1,j,c} - \bar{z}_k^{1,u}\right)\left(z_k^{1,j,c} - \bar{z}_k^{1,u}\right)^{\mathrm{T}} \\ &= \sum_{j=1}^{n_k^1} \left(z_k^{1,j} - \bar{z}_k^{1,u}\right)\left(z_k^{1,j} - \bar{z}_k^{1,u}\right)^{\mathrm{T}} \\ &\quad + \frac{n_k^1\left(1 - c_{D_k^1}\right)}{c_{D_k^1}} \bar{z}_k^{1,u}\left(\bar{z}_k^{1,u}\right)^{\mathrm{T}} + \sum_{j=1}^{n_k^{1,c}} z_k^{1,j,c}\left(z_k^{1,j,c}\right)^{\mathrm{T}} \\ &\quad - \left(\sum_{j=1}^{n_k^{1,c}} z_k^{1,j,c}\right)\left(\bar{z}_k^{1,u}\right)^{\mathrm{T}} - \bar{z}_k^{1,u}\left(\sum_{j=1}^{n_k^{1,c}} z_k^{1,j,c}\right)^{\mathrm{T}}. \end{aligned} \tag{22b}$$

For practicality, instead of constructing $n_k^{1,c}$ (which may not be an integer) pseudo measurements, we can further express $\sum_{j=1}^{n_k^{1,c}} z_k^{1,j,c}$ and $\sum_{j=1}^{n_k^{1,c}} z_k^{1,j,c}\left(z_k^{1,j,c}\right)^{\mathrm{T}}$ using the analytic mean and the second moment of $z_k^{1,j,c}$ as

$$\begin{aligned} \sum_{j=1}^{n_k^{1,c}} z_k^{1,j,c} &\approx n_k^{1,c}\mathbb{E}\left[z_k^{1,j,c}\right] \\ &= \frac{n_k^1\left(1 - c_{D_k^1}\right)}{c_{D_k^1}} \left(M(\phi_k)E(X_k)\mathbb{E}\left[y_k^{1,j,c}\right] + Hx_k\right) \end{aligned} \tag{23a}$$

$$\begin{aligned} \sum_{j=1}^{n_k^{1,c}} z_k^{1,j,c}\left(z_k^{1,j,c}\right)^{\mathrm{T}} &\approx n_k^{1,c}\mathbb{E}\left[z_k^{1,j,c}\left(z_k^{1,j,c}\right)^{\mathrm{T}}\right] \\ &= \frac{n_k^1\left(1 - c_{D_k^1}\right)}{c_{D_k^1}} \Bigg(\mathbb{E}\left[z_k^{1,j,c}\right]\mathbb{E}\left[z_k^{1,j,c}\right]^{\mathrm{T}} \\ &\quad + M(\phi_k)E(X_k)\left(\mathrm{Cov}\left[y_k^{1,j,c}\right] + R_k^1\right)E(X_k)^{\mathrm{T}}M(\phi_k)^{\mathrm{T}}\Bigg) \end{aligned} \tag{23b}$$

where $\mathbb{E}[y_k^{1,j,c}]$ and $\mathrm{Cov}[y_k^{1,j,c}]$ are, respectively, the mean and the covariance of $\mathcal{TN}(y_k^{1,j,c}; 0_{2\times1}, \rho^1 \mathrm{I}_2, D_k^{1,c})$; see Appendix E for their explicit expressions. Equations (22) and (23) can be interpreted in a way that we make use of the mean and the second moment of a HTG distribution to correct the biases introduced in directly using the centroid measurement and the measurement spread to perform the random matrix measurement update.

Given SS measurements Z_k^1 and the predicted GIW parameters $\left(m_{k|k-1}, P_{k|k-1}, v_{k|k-1}, V_{k|k-1}\right)$, at the tth iteration the posterior Gaussian parameters $\left(m_{k|k}^{\mathrm{SS},t}, P_{k|k}^{\mathrm{SS},t}\right)$ are

$$m_{k|k}^{\mathrm{SS},t} = m_{k|k-1} - K_{k|k-1}^1\left(\bar{z}_k^{1,u,t-1} - Hm_{k|k-1}\right), \tag{24a}$$

$$P_{k|k}^{\mathrm{SS},t} = P_{k|k-1} - K_{k|k-1}^1 HP_{k|k-1}, \tag{24b}$$

$$K_{k|k-1}^1 = P_{k|k-1}H^{\mathrm{T}}\left(S_{k|k-1}^1\right)^{-1}, \tag{24c}$$

$$S_{k|k-1}^1 = HP_{k|k-1}H^{\mathrm{T}} + \hat{Y}_{k|k-1}^1, \tag{24d}$$

$$\hat{Y}_{k|k-1}^1 = \frac{c_{D_k^1}}{n_k^1}\left(\rho^1\hat{X}_{k|k}^{t-1} + R_k^{1,g,t-1}\right) \tag{24e}$$

and the posterior IW parameters $\left(v^{\text{SS},t}_{k|k}, V^{\text{SS},t}_{k|k}\right)$ are

$$v^{\text{SS},t}_{k|k} = v_{k|k-1} + \frac{n^1_k}{c_{D^1_k}}, \tag{25a}$$

$$\hat{V} = V_{k|k-1} + Z^1_{k|k-1} + L^1_{k|k-1}, \tag{25b}$$

$$V^{\text{SS},t}_{k|k} = M(\hat{\phi}^t_{k|k})^{\text{T}} E(\hat{V})^2 M(\hat{\phi}^t_{k|k}), \tag{25c}$$

$$\begin{aligned} Z^1_{k|k-1} = {} & \hat{X}^{\frac{1}{2}}_{k|k-1} \left(\frac{n^1_k}{c_{D^1_k}} \hat{Y}^1_{k|k-1}\right)^{-\frac{1}{2}} \bar{Z}^{1,u,t-1}_k \\ & \times \left(\frac{n^1_k}{c_{D^1_k}} \hat{Y}^1_{k|k-1}\right)^{-\frac{\text{T}}{2}} \hat{X}^{\frac{\text{T}}{2}}_{k|k-1}, \end{aligned} \tag{25d}$$

$$L^1_{k|k-1} = \hat{X}^{\frac{1}{2}}_{k|k-1} \left(S^1_{k|k-1}\right)^{-\frac{1}{2}} N^1_{k|k-1} \left(S^1_{k|k-1}\right)^{-\frac{\text{T}}{2}} \hat{X}^{\frac{\text{T}}{2}}_{k|k-1}, \tag{25e}$$

$$N^1_{k|k-1} = \left(\bar{z}^{1,u,t-1}_k - Hm_{k|k-1}\right)\left(\bar{z}^{1,u,t-1}_k - Hm_{k|k-1}\right)^{\text{T}} \tag{25f}$$

where variables with superscript $t-1$ are computed using the updated state estimate $\hat{\xi}^{t-1}_{k|k}$ at the $(t-1)$th iteration.

Several differences between the FFK update [10] and the update equations (24), (25) are noted below. First, $n^1_k / c_{D^1_k}$ is regarded as the number of measurements instead of n^1_k; the former can be interpreted as the approximate number of measurements when the spatial density is not affected by self-occlusion. Second, the assumption that $X_k \approx \hat{X}_{k|k-1}$ in the FFK update is relaxed by $X_k \approx \hat{X}^{t-1}_{k|k}$, which could be more accurate when there is no accurate prior information about the object size. Last, equation (25c) can be interpreted in a way that we ignore the orientation information contained in X_k and enforce it to be the object heading in the kinematic state[4]. Empirical results show that this can lead to improved estimation performance. The reasons here are two fold. First, the object orientation is assumed to be aligned with the object heading. Second, the object kinematic state estimate is usually more accurate than the measure of orientation obtained via the eigen-decomposition of the extent matrix, even when there are few measurements.

B. Multi-Sensor Update

We proceed to present the MS extension of the proposed random matrix update method. Similar to the SS update for HTG spatial density, for each sensor we augment the sensor measurements Z^s_k with a set of pseudo measurements $Z^{s,c}_k$ such that the set of measurements $Z^{s,u}_k = Z^s_k \cup Z^{s,c}_k$ can be regarded as Gaussian distributed. The MS likelihood for the augmented sensor measurements can be written as

$$\begin{aligned} p(Z^u_k|\xi_k) = {} & \prod_{s=1}^{S} \prod_{z^{s,u}_k \in Z^{s,u}_k} \mathcal{N}\left(z^{s,u}_k; Hx_k, \rho^s X_k + R^{s,g}_k\right) \\ \propto {} & \prod_{s=1}^{S} \mathcal{N}\left(\bar{z}^{s,u}_k; Hx_k, \frac{\rho^s_k X_k + R^{s,g}_k}{n^s_k}\right) \\ & \times \prod_{s=1}^{S} |\rho^s_k X_k + R^{s,g}_k|^{-\frac{|Z^{s,u}_k|-1}{2}} \\ & \times \operatorname{etr}\left(-\frac{1}{2}\bar{Z}^{s,u}_k \left(\rho^s_k X_k + R^{s,g}_k\right)^{-1}\right). \end{aligned} \tag{26}$$

The derivation of the second step of (26) can be found in e.g., [9, Appendix C] or [32, Appendix A].

We can further obtain the approximate $\bar{z}^{s,u}_k$ and $\bar{Z}^{s,u}_k$ by making use of the analytic mean and the second moment of the HTG distributed random variable $z^{s,c}_k$, cf. (23). With (26), the MS generalization of the SS update (24) and (25) follows the MSFFK approach [27], which makes use of the centroid measurements and measurement spreads for the individual sensors. The main difference here is that the centroid measurement $\bar{z}^s_k$ and the measurement spread $\bar{Z}^s_k$ of received sensor measurements are replaced with (22), which are computed using the means, variances and normalization factors of different HTG distributions.

Given MS measurements Z_k and the predicted GIW parameters $\left(m_{k|k-1}, P_{k|k-1}, v_{k|k-1}, V_{k|k-1}\right)$, at the tth iteration the posterior Gaussian parameters $\left(m^{\text{MS},t}_{k|k}, P^{\text{MS},t}_{k|k}\right)$ are

$$m^{\text{MS},t}_{k|k} = m_{k|k-1} + K^f_{k|k-1}\left(\bar{z}^f_{k|k-1} - Hm_{k|k-1}\right), \tag{27a}$$

$$P^{\text{MS},t}_{k|k} = P_{k|k-1} - K^f_{k|k-1} H P_{k|k-1}, \tag{27b}$$

$$K^f_{k|k-1} = P_{k|k-1} H^{\text{T}} \left(S^f_{k|k-1}\right)^{-1}, \tag{27c}$$

$$S^f_{k|k-1} = H P_{k|k-1} H^{\text{T}} + \hat{Y}^f_{k|k-1}, \tag{27d}$$

$$\bar{z}^f_{k|k-1} = \hat{Y}^f_{k|k-1} \sum_{s=1}^{S} \left(\hat{Y}^s_{k|k-1}\right)^{-1} \bar{z}^{s,u,t-1}_k, \tag{27e}$$

$$\hat{Y}^f_{k|k-1} = \left(\sum_{s=1}^{S} \left(\hat{Y}^s_{k|k-1}\right)^{-1}\right)^{-1}, \tag{27f}$$

$$\hat{Y}^s_{k|k-1} = \frac{c_{D^s_k}}{n^s_k}\left(\rho^s \hat{X}^{t-1}_{k|k} + R^{s,g,t-1}_k\right), \tag{27g}$$

$$R^{s,g,t-1}_k = M(\hat{\phi}^{t-1}_{k|k}) E(\hat{X}^{t-1}_{k|k}) R^s_k E(\hat{X}^{t-1}_{k|k})^{\text{T}} M(\hat{\phi}^{t-1}_{k|k})^{\text{T}}, \tag{27h}$$

and the posterior IW parameters $\left(v^{\text{MS},t}_{k|k}, V^{\text{MS},t}_{k|k}\right)$ are

$$v^{\text{MS},t}_{k|k} = v_{k|k-1} + \sum_{s=1}^{S} \frac{n^s_k}{c_{D^s_k}}, \tag{28a}$$

$$\hat{V} = V_{k|k-1} + \sum_{s=1}^{S} Z^s_{k|k-1} + L^f_{k|k-1}, \tag{28b}$$

$$V^{\text{MS},t}_{k|k} = M(\hat{\phi}^t_{k|k})^{\text{T}} E(\hat{V})^2 M(\hat{\phi}^t_{k|k}), \tag{28c}$$

$$\begin{aligned} Z^s_{k|k-1} = {} & \hat{X}^{\frac{1}{2}}_{k|k-1} \left(\frac{n^s_k}{c_{D^s_k}} \hat{Y}^s_{k|k-1}\right)^{-\frac{1}{2}} \bar{Z}^{s,u,t-1}_k \\ & \times \left(\frac{n^s_k}{c_{D^s_k}} \hat{Y}^s_{k|k-1}\right)^{-\frac{\text{T}}{2}} \hat{X}^{\frac{\text{T}}{2}}_{k|k-1}, \end{aligned} \tag{28d}$$

$$L^f_{k|k-1} = \hat{X}^{\frac{1}{2}}_{k|k-1} \left(S^f_{k|k-1}\right)^{-\frac{1}{2}} N^f_{k|k-1} \left(S^f_{k|k-1}\right)^{-\frac{\text{T}}{2}} \hat{X}^{\frac{\text{T}}{2}}_{k|k-1}, \tag{28e}$$

[4] It is also possible to consider the de-coupling on matrices $Z^1_{k|k-1}$ and $L^1_{k|k-1}$ before matrix $\hat{V}$ is computed.

Algorithm 1 Multi-Sensor GIW Update for HTG Model

```
Input: Measurements from $S$ sensors $Z_k = \{Z_k^s\}_{s=1}^{S}$, a set of $N_h$ learned HTG models $\{(D^i, R^i, \rho^i)\}_{i=1}^{N_h}$, predicted GIW parameters $(m_{k|k-1}, P_{k|k-1}, v_{k|k-1}, V_{k|k-1})$.
Output: Posterior GIW parameters $(m_{k|k}, P_{k|k}, v_{k|k}, V_{k|k})$.
Initialize $\hat{\xi}_{k|k}^0 = \hat{\xi}_{k|k-1}$.
while convergence is not reached do
    for $s = 1$ to $S$ do
        Compute object kinematic state $\hat{\xi}_{k|k}^t$ in the frame of the $s$th sensor.
        Compute approximate aspect angle (14).
        Select a learned HTG model based on aspect angle and convert it to the global frame.
        Compute $\bar{z}_k^{s,u}$ (22a) and $\bar{Z}_k^{s,u}$ (22b) using approximation (23).
        Compute $\hat{Y}_{k|k-1}^s$ (27g) and $\hat{Z}_{k|k-1}^s$ (28d).
    end for
    Compute $(m_{k|k}^t, P_{k|k}^t)$ (27) and $(v_{k|k}^t, V_{k|k}^t)$ (28).
end while
Set $(m_{k|k}^t, P_{k|k}^t, v_{k|k}^t, V_{k|k}^t)$ as $(m_{k|k}, P_{k|k}, v_{k|k}, V_{k|k})$.
```

$$N_{k|k-1}^f = \left(\bar{z}_{k|k-1}^f - Hm_{k|k-1}\right)\left(\bar{z}_{k|k-1}^f - Hm_{k|k-1}\right)^{\mathrm{T}} \tag{28f}$$

where $\bar{z}_k^{s,u,t-1}$ (cf. (22a)) and $\bar{Z}_k^{s,u,t-1}$ (cf. (22b)) are, respectively, the approximate centroid measurement and measurement spread for the sth sensor computed using the updated state estimate $\hat{\xi}_{k|k}^{t-1}$ at the $(t-1)$th iteration.

The approximation errors introduced by (23) and (22) for the individual sensors decrease as the number of sensor measurements n_k^s increases. Nevertheless, the experiment results in Section VI show that the proposed MS update for HTG model has promising estimation performance even when there are few measurements. For the extent matrix update, we have sums over the approximate spreads for the individual sensors $\sum_{s=1}^{S} Z_{k|k-1}^s$ and the spread $N_{k|k-1}^f$ around the approximate centroid for all sensors. It is also possible to compute the approximate spread for all measurements from all sensors, i.e., $\sum_{s=1}^{S} \bar{Z}_k^{s,u}$. This will result in a slightly different implementation similar to the fusion approximation in [31].

When $S = 1$, the MS update (27), (28) reduces to the SS update (24), (25). The pseudo code for the complete MS GIW update for HTG model is given in Algorithm 1. Note that the means, variances, and normalization factors of the learned truncated Gaussian distributions can be precomputed before running the algorithm to avoid repetitive computations.

C. Prediction

We briefly review the state prediction for factorized GIW density representation for completeness. The kinematic state transition density is

$$p(x_k|x_{k-1}) = \mathcal{N}\left(x_k; f_{k,k-1}(x_{k-1}), Q_k^{\mathrm{KS}}\right) \tag{29}$$

where $f_{k,k-1}(x_{k-1})$ is a function that describes the (non-linear) motion of the object and Q_k^{KS} is the motion process noise covariance. The predicted Gaussian parameters $(m_{k|k-1}, P_{k|k-1})$ can be obtained by the EKF prediction

$$m_{k|k-1} = f_{k,k-1}(m_{k-1|k-1}), \tag{30a}$$

$$P_{k|k-1} = F_{k,k-1}P_{k-1|k-1}F_{k,k-1}^{\mathrm{T}} + Q_k^{\mathrm{KS}}, \tag{30b}$$

$$F_{k,k-1} = \nabla_{x_{k-1}} f_{k,k-1}(x_{k-1})|_{x_{k-1}=m_{k-1|k-1}} \tag{30c}$$

where $F_{k,k-1}$ is the Jacobian matrix.

By modeling the extent matrix state transition density as a non-central IW distribution [30, Def. 3.5.2], parameterized as

$$p(X_k|X_{k-1}) = \mathcal{IW}^{\mathrm{nc}}\left(X_k; v_k, A_kQ_k^{\mathrm{ES}}A_k^{\mathrm{T}}, A_kQ_k^{\mathrm{ES}}X_{k-1}^{-1}A_k^{-1}\right) \tag{31}$$

where A_k is the nonsingular transition matrix and Q_k^{ES} is the extent noise matrix, a recent work [14] shows that the resulting predicted density for an IW extent matrix posterior is also an IW distribution without approximation. The predicted IW parameters $(v_{k|k-1}, V_{k|k-1})$ are

$$v_{k|k-1} = 6 + (v_{k-1|k-1} - 6)\left|\mathrm{I}_2 + Q_k^{\mathrm{ES}}V_{k-1|k-1}\right|^{-\frac{1}{2}}, \tag{32a}$$

$$V_{k-1|k-1} = A_kV_{k-1|k-1}\left(\mathrm{I}_2 + Q_k^{\mathrm{ES}}V_{k-1|k-1}\right)^{-1}A_k^{\mathrm{T}}. \tag{32b}$$

With the assumptions that the object orientation is aligned with the object heading and that the noise matrix Q_k^{ES} has no impact on the extent matrix, one can set $Q_k^{\mathrm{ES}} = V_{k-1|k-1}^{-1}/\alpha_k$ and $A_k = M(\omega_{k-1}T_s)$ with T_s denoting the sampling period and ω_{k-1} denoting the turn rate at time step $k-1$, such that

$$v_{k|k-1} = 6 + \frac{\alpha_k}{1+\alpha_k}(v_{k-1|k-1} - 6), \tag{33a}$$

$$V_{k-1|k-1} = \frac{\alpha_k}{1+\alpha_k}M(\omega_{k-1}T_s)V_{k-1|k-1}M(\omega_{k-1}T_s)^{\mathrm{T}} \tag{33b}$$

where α_k is a tuning parameter controling the noise level. The larger α_k is, the more confident the prediction (33) becomes. We note that (33) resembles the heuristic extent prediction for a Wishart extent transition density [10].

V. Rao-Blackwellized Particle Filter for Hierarchical Truncated Gaussian Model

In this section, we extend the random matrix approach [27] based on a RBP representation to HTG spatial density. The RBP representation of the object state density assumes a Gaussian distribution for the kinematic state x_k and a particle approximation of the extent state X_k whose orientation is determined by the object heading contained in x_k[5]. By having a particle representation of the object extent state and limiting the sampling space to object length and width, we aim at obtaining an implementation that offers a good trade-off between computational complexity and estimation performance. Also, the RBP filter implementation can be regarded as a fair benchmark for the implementation based on factorized GIW density representation.

For a RBP representation of the state density, the predicted and posterior densities are, respectively, approximated as

$$p(\xi_k|Z_{1:k-1}) \approx p(x_k|Z_{1:k-1})p(X_k|Z_{1:k-1})$$

[5] It is also possible to represent each particle as a vector consisting of the object length and width. We stick to the extent matrix representation here merely for the purpose of simplifying notations. Note that the practical implementations for both representations are the same.

$$= \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}) \sum_{i=1}^{N_p} w_{k-1}^{(i)} \delta_{X_k}(X_k^{(i)}), \tag{34a}$$

$$p(\xi_k|Z_{1:k}) \approx p(x_k|Z_{1:k}) p(X_k|Z_{1:k})$$
$$= \mathcal{N}(x_k; m_{k|k}, P_{k|k}) \sum_{i=1}^{N_p} w_k^{(i)} \delta_{X_k}(X_k^{(i)}) \tag{34b}$$

where $w_k^{(i)}$ is the weight of particle i, N_p is the number of particles and $\delta_X(Y)$ is the Dirac delta function. For particle representation of X_k, the expected extent is $\hat{X}_k = \sum_{i=1}^{N_p} X_k^{(i)} w_k^{(i)}$. In the RBP state update presented in [27], the kinematic state is approximated by its prediction to split the state update into two parts: Gaussian kinematic update and particle extent update. Similar to the GIW state update for HTG spatial density, an iterative update method is used here. Because the kinematic update is the same as (24), we focus on the extent update.

We proceed to present the measurement likelihood function and the importance distributions used to draw samples in the sequential importance resampling. At the tth iteration, we approximate the kinematic state $x_k \approx m_{k|k}^t$, and the MS measurement likelihood (8) can be written as

$$p(Z_k|\xi_k) = \prod_{s=1}^{S} \mathcal{L}_k^s \left(Z_k^s | X_k, m_{k|k}^t \right) \tag{35}$$

where $\mathcal{L}_k^s(Z_k^s|\cdot)$ is shorthand notation for the measurement likelihood function for the sth sensor conditioned on the aspect angle (14); its explicit expression can be derived following (8) and (12). As the object orientation is assumed to be aligned with the object heading, it is sufficient to first draw samples of X_k in the object frame, i.e., $E(X_k)^2$, from an assumed density. We can then transform the generated samples to the global frame to obtain the particles $X_k^{(i),t}$. Therefore the importance distribution can be chosen as

$$q(\cdot|X_{k-1}^{(i)}, Z_k) = M(\hat{\phi}_{k|k}^t) \left(E(X_{k-1}^{(i)}) + \sigma_k \right) \tag{36a}$$
$$\times \left(E(X_{k-1}^{(i)}) + \sigma_k \right)^{\mathrm{T}} M(\hat{\phi}_{k|k}^t)^{\mathrm{T}},$$
$$\sigma_k \sim \mathcal{N}\left(0_{2\times 1}, \operatorname{diag}\left([\sigma_{l_1}^2/4, \sigma_{l_2}^2/4] \right) \right) \tag{36b}$$

where σ_{l_1} and σ_{l_2} are the standard deviations for the major and minor axes, respectively. Note that the orientation of $X_k^{(i),t}$ is determined by the object heading estimate $\hat{\phi}_{k|k}^t$ rather than from sampling, compared to the RBP state update proposed in [27] for Gaussian spatial density.

The main steps of the sequential importance resampling at the tth iteration can be summarized as follows. First, N_p new particles are generated using (36). Second, new particle weights are computed according to

$$w_k^{(i),t} \propto w_{k-1}^{(i)} \prod_{s=1}^{S} \mathcal{L}_k^s \left(Z_k^s | X_k^{(i),t}, m_{k|k}^t \right), \tag{37}$$

and normalized to sum to one. Lastly, resampling is performed if the effective number of particles [33]

$$N_p^{\text{eff}} \approx \frac{1}{\sum_{i=1}^{N_p} \left(w_k^{(i),t} \right)^2} \tag{38}$$

Algorithm 2 Multi-Sensor RBP Update for HTG Model

```
Input: Measurements from $S$ sensors $Z_k = \{Z_k^s\}_{s=1}^S$, a set of $N_h$ learned HTG models $\{(D^i, R^i, \rho^i)\}_{i=1}^{N_h}$, predicted RBP parameters $(m_{k|k-1}, P_{k|k-1}, w_{k-1}^{(i)}, X_{k-1}^{(i)})$.
Output: Posterior RBP parameters $(m_{k|k}, P_{k|k}, w_k^{(i)}, X_k^{(i)})$.
Initialize $\hat{\xi}_{k|k}^0 = \hat{\xi}_{k|k-1}$.
while convergence is not reached do
    for $s = 1$ to $S$ do
        Compute object kinematic state $\hat{\xi}_{k|k}^t$ in the frame of the $s$th sensor.
        Compute approximate aspect angle (14).
        Select a learned HTG model based on aspect angle and convert it to the global frame.
        Compute $\bar{z}_k^{s,u}$ (22a) using approximation (23a).
        Compute $\hat{Y}_{k|k-1}^s$ (27g).
    end for
    Compute $(m_{k|k}^t, P_{k|k}^t)$ (27).
    For $i = 1, \ldots, N_p$, obtain $X_k^{(i),t} \sim q(\cdot|X_{k-1}^{(i)}, Z_k)$ (36).
    Compute new particle weights $w_k^{(i),t}$ (37).
    Normalize weights $\sum_{i=1}^{N_p} w_k^{(i),t} = 1$.
    Compute effective number of particles $N_p^{\text{eff}}$ (38).
    if $N_p^{\text{eff}}$ is significantly lower than $N_p$ then
        Resample $\left\{ w_k^{(i),t}, X_k^{(i),t} \right\}_{i=1}^{N_p}$.
    end if
end while
Set $(m_{k|k}^t, P_{k|k}^t, w_k^{(i),t}, X_k^{(i),t})$ as $(m_{k|k}, P_{k|k}, w_k^{(i)}, X_k^{(i)})$.
```

is significantly less than N_p, e.g., $N_p^{\text{eff}} < N_p/4$, to avoid the particle degeneracy problem. Afterward, the extent estimate $\hat{X}_{k|k}^t$ can be used for Gaussian kinematic update at the next iteration. The pseudo code for the complete MS RBP update for HTG model is given in Algorithm 2. Note that, though the state update is iterative, we only need to draw N_p samples from the object frame, i.e., $E(X_{k-1}^{(i)}) + \sigma_k$ at the first iteration to avoid redundant computations.

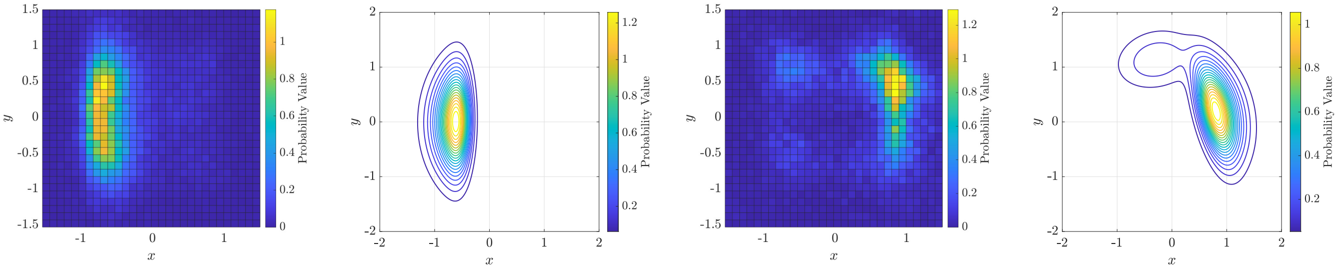
The kinematic prediction is the same as (30), (33). With the assumption that the object extent is time-invariant, the object extent prediction is given by $\hat{X}_{k|k-1} = \hat{X}_{k-1|k-1}$.

VI. Performance Evaluation

In this section, we first describe the HTG model learning using the nuScenes dataset [20], a real-world public automotive radar dataset. Then we present the simulation results on synthetic data generated using the learned HTG models. Finally, experimental validation on the nuScenes dataset is presented.

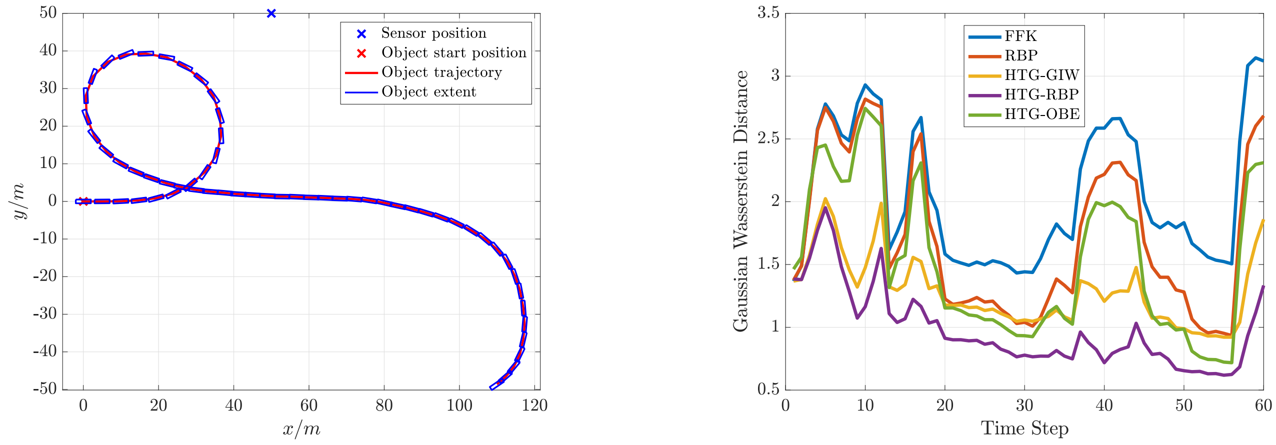
A. Data Preprocessing and Model Training

The large-scale public nuScenes dataset contains multi-modal measurements from a full sensor suite including LIDAR, camera, global positioning system (GPS), and radar. Particularly, annotated keyframes are provided at a sample rate of 2-Hz for 1000 scenes of 20-second duration. The ego vehicle is mounted with five 77-GHz frequency modulated continuous wave (FMCW) long-range radars at the four corners and the front bumper of the ego vehicle. Each radar sensor has a 13-Hz capture frequency and a maximum range of 250 meters.



(a) $x' \in [-\pi/8, \pi/8)$ (b) $x' \in [-\pi/8, \pi/8)$ (c) $x' \in [5\pi/8, 7\pi/8)$ (d) $x' \in [5\pi/8, 7\pi/8)$

Fig. 5. Histograms of radar detection points and probability density functions of learned HTG models, conditioned on different aspect angle ranges.



(a) (b)

Fig. 6. Simulated scenario: (a) ground truth with the simulated trajectory and (b) average GWDs over time with synthetic measurements.

To accumulate radar detection points for vehicles, we first extracted the set of objects labeled as car, and then for each object we extracted all radar points within its annotated bounding box with a scaling ratio of 1.5. This is to account for the spread of the measurements around object edges. To reduce the number of clutter points in the training dataset, we utilized the annotation information to only include detection points with unambiguous radial velocity and probability of being an artifact less than 25%. After the clutter removal, each radar point was transformed to the unit frame using the annotated object size and concatenated with its corresponding aspect angle (14). This resulted in a training dataset with a total of 175760 detection points.

A set of 8 HTG models were trained based on subsets of radar points conditioned on the aspect angle χ of (14) in the equally spaced intervals $[-\pi, \pi]$ in the unit frame [23]. Two of the 8 learned HTG models are shown in Fig. 5.

B. Performance Evaluation on Synthetic Data

We consider a scenario of a rectangular object (4-m long and 1-m wide) maneuvering with a nonlinear motion for 60 time steps. The object is represented by an ellipse with the major and minor axes matching with the length and, respectively, width of the object. The sensor position is fixed with its orientation pointing towards the object center. The tracking scenario is illustrated in Fig. 6 (a). The number of measurements at each time step is drawn from a Poisson distribution with a mean of 12. The measurements are randomly generated using the offline trained HTG models conditioned on the aspect angle with a sampling frequency of 2-Hz. The kinematic vector is given as $x_k = [p_{k,x}, p_{k,y}, \nu_k, \phi_k, \omega_k]^{\mathrm{T}}$ where ν_k is the object velocity, and the framed turn motion model is used with standard deviations of polar and angular acceleration noises $\sigma_{\dot{\nu}} = 0.5$ and $\sigma_{\dot{\omega}} = 0.1$, respectively. The explicit expressions for the transition model $f_{k,k-1}(\cdot)$ and process noise covariance matrix Q_k^{KS} can be found in [34].

For tracking performance comparison, 5 EOT algorithms are considered:

1) random matrix approach with FFK update [10];
2) random matrix approach with RBP update [27];
3) random matrix approach for HTG model with GIW density and online truncation bounds estimation (referred to as HTG-OBE) [24], [25];
4) random matrix approach for HTG model with GIW density and offline learned structural parameters (referred to as HTG-GIW) presented in Section IV;
5) random matrix approach for HTG model with RBP density representation and offline learned structural parameters (referred to as HTG-RBP) presented in Section V.

All the algorithms are initialized using the true kinematic state and extent matrix $X_0 = \mathrm{I}_2$. To implement with GIW density, the initial degree of freedom in the IW density is chosen as

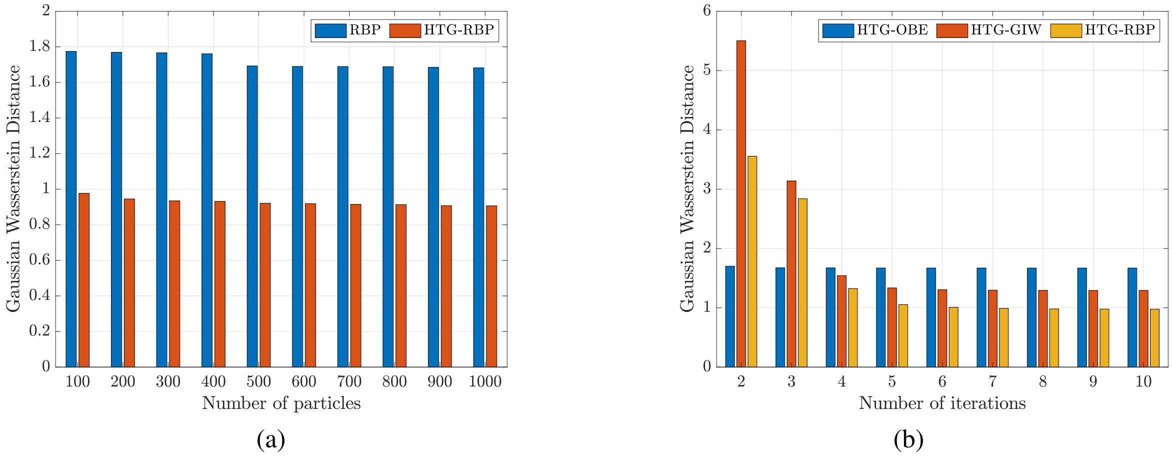


Fig. 7. Average GWD error per time step versus (a) the number of particles and (b) the number of iterations used in the iterative state update.

$v_0 = 1000$, and the tuning parameter in extent prediction (33) is set to $\alpha = +\infty$ to model time-invariant object extent [9]. For implementations using RBP, the number of particles is set to 100 and, in the importance distribution (36), the standard deviations for the major and minor axes are $\sigma_{l_1} = 0.4$ and $\sigma_{l_2} = 0.15$, respectively. For implementations not using offline learned structural parameters, the scaling factor is fixed to $\rho = 0.25$ and the measurement noise covariance matrix is $R^g = \text{diag}([0.04, 0.04])$. Finally, for HTG implementations, the number of iterations used for iterative state update is 10.

The estimation performance is evaluated using the Gaussian Wasserstein distance (GWD) metric [35]. The GWD between two state estimates $\xi_1 = (x_1, X_1)$ and $\xi_2 = (x_2, X_2)$ is [36]

$$W_2(\xi_1, \xi_2)^2 = ||x_1 - x_2||_2^2 + \text{Tr}\{X_1 + X_2 - 2(X_1 X_2)^{\frac{1}{2}}\}. \quad (39)$$

The GWDs over time, averaged over 1000 Monte Carlo trials, are depicted in Fig. 6 (b). It is seen that FFK presents the worst estimation performance followed by RBP. As both FFK and RBP are algorithms developed for Gaussian spatial densities, they suffer from measurement model mismatch. Within the HTG approaches, HTG-GIW outperforms HTG-OBE because the measurements are generated using the trained HTG models. HTG-RBP shows improved estimation performance over HTG-GIW at the expense of higher computational complexity.

The performance sensitivity of RBP and HTG-RBP with respect to the number of particles is shown in Fig. 7 (a). The results show that the estimation performance improvement of HTG-RBP is less than 10% by increasing the number of particles from 100 to 1000. Therefore the number of particles of 100 may be advisable in terms of computational efficiency.

Similarly, the performance sensitivity of HTG-OBE, HTG-GIW and HTG-RBP with respect to the number of iterations used in the iterative state update is shown in Fig. 7 (b). The results show that, on average, HTG-OBE converges within 3 iterations, while HTG-GIW and HTG-RBP take up to 6 iterations for convergence. The reason that HTG-OBE converges faster than HTG-GIW and HTG-RBP is because the HTG model used in HTG-OBE is defined in the global frame, independent of the object size. When applied in practical applications, the number of iterations of 10 may be advisable for HTG-GIW and HTG-RBP.

C. Performance Evaluation on Experimental Data

To further evaluate performance using the real-world nuScenes dataset, we consider 1) HTG-OBE [24], [25]; 2) HTG-GIW presented in Section IV; 3) HTG-RBP presented in Section V; 4) probability multiple hypothesis tracker (PMHT) with offline learned GMM (referred to as GMM-PMHT) [23]; and 5) Cartesian B-spline model-based approach for vehicle tracking [8]. We exclude the conventional random matrix approaches with FFK and RBP updates as they were reported to give worse performance than the GMM-PMHT and B-spline approaches for the nuScene dataset [23]. In GMM-PMHT, the measurement spatial densities conditioned on different aspect angles in the unit frame are modeled as Gaussian mixtures, and an EM formulation of the PMHT for single time steps is used for state update. The B-spline model allows for a good approximation of the object contour features such as the rounded corners of cars. Compared to random matrix-based approaches, the object extent in the GMM-PMHT and B-spline model is explicitly represented using object length and width in an augmented kinematic state.

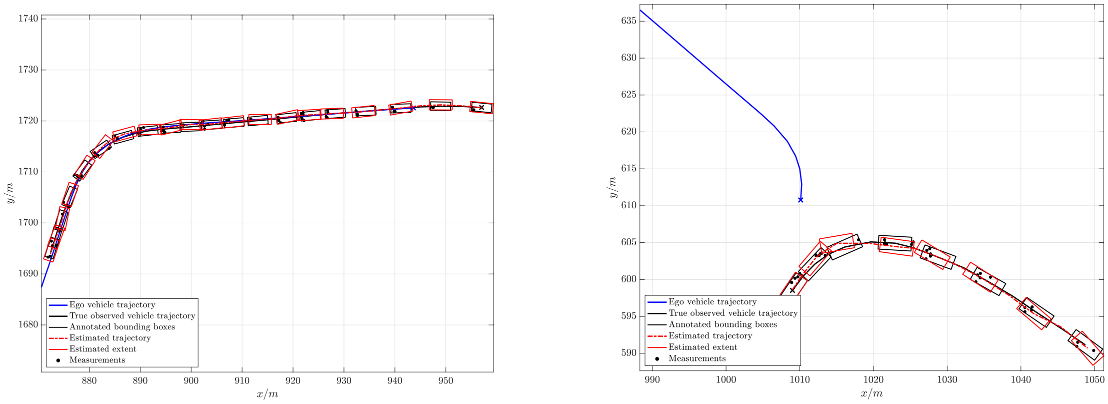
For the nuScenes dataset, we only consider annotated trajectories of cars that are continuously covered by at least one radar point for a duration no less than 10 time steps. This results in a total of 300 selected trajectories. It is noted that the radar measurements in the nuScenes dataset are sparse: the average number of radar measurements per sensor per time step is less than 2. The coordinate turn motion model is used. For each trajectory, we use the standard deviations of the true object polar and angular accelerations as the standard deviations of polar acceleration noises $\sigma_{\dot{\nu}}$ and $\sigma_{\dot{\omega}}$, respectively. For all considered algorithms, the initial kinematic state is set to the same as the first annotated object state and the initial extent state is determined by the average size of objects (4.5-m long and 2-m wide).

For HTG implementations, the same parameter setting was used as in the simulation study. For the GMM-PMHT and B-spline model, the EKF is used for state prediction and update,

TABLE I

Means and Standard Deviations of Estimation Errors per Time Step, Averaged Over 300 Selected Trajectories.

	GWD		Localization (m)		Heading (rad)		Length (m)		Width (m)		Execution time (s)	
Algorithm	mean	std	mean	std	mean	std	mean	std	mean	std	mean	std
HTG-OBE	1.3531	0.4164	1.2659	0.3961	0.0661	0.1463	0.3311	0.2836	**0.1257**	0.0957	0.0191	0.0084
HTG-GIW	1.0072	0.5446	0.9211	0.5178	0.0638	0.0686	0.3372	0.2761	0.1271	0.0938	0.0054	0.0031
HTG-RBP	**1.0056**	0.5401	0.9183	0.5131	**0.0637**	0.0676	0.3346	0.2819	0.1271	0.0965	0.1252	0.0282
GMM-PMHT	1.0409	1.1975	**0.7503**	0.6434	0.2721	0.6661	**0.3040**	0.2719	0.5811	0.4362	0.0259	0.0104
B-spline	2.5164	2.5919	2.0832	2.0071	0.4136	0.7766	0.5572	0.3750	0.4140	0.4196	**0.0015**	0.0015



(a) (b)

Fig. 8. Examples of maneuvering vehicle tracking using HTG-GIW for two nuScenes trajectories illustrated at every two time steps.

the initial covariance matrix on object length and width is $\mathrm{diag}([0.16, 0.225])$, the additional process noise on the object size is zero, and the sensor fusion is performed by sequentially processing the measurements from each sensor. For the GMM-PMHT, a set of eight GMMs were trained based on the same division of radar points used in the HTG model training[6], and the number of EM iterations used in state update is also 10. Finally, for the B-spline model and HTG-OBE, the measurement noise covariance matrix is $R^g = \mathrm{diag}([0.25, 0.25])$.

The experiment results[7] on 300 selected trajectories are summarized in Table I and two examples of maneuvering vehicle tracking using HTG-GIW are illustrated in Fig. 8. The overall estimation performance is assessed using the GWD metric. For the GMM-PMHT and B-spline model, the extent matrix can be reconstructed using

$$X_k = M(\phi_k)\,\mathrm{diag}([l_1^2/4, l_2^2/4])M(\phi_k)^{\mathrm{T}} \tag{40}$$

where l_1 and l_2 are the object length and width, respectively.

Table I shows clear performance improvements in terms of mean GWD error from the HTG-OBE and B-spline approaches that do not explore offline training datasets to the learning-based approaches (HTG-GIW, HTG-RBP and GMM-PMHT). Among the learning-based approaches, HTG-GIW is about 5 times faster than GMM-PMHT and about 20 times faster than HTG-RBP. HTG-GIW uses the approximate measurement mean and spread for state update, and therefore its computational complexity scales well with the number of measurements. The B-spline model has the lowest computational cost but also the worst estimation performance. The B-spline model in [8] neglects the sensor state and uses a greedy method to associate measurements to object contour, and therefore it may not work well when there are few measurements or when measurements do not always appear around the object contour. GMM-PMHT has the best localization performance, but it also gives less accurate estimates to the object heading and width when compared to HTG implementations.

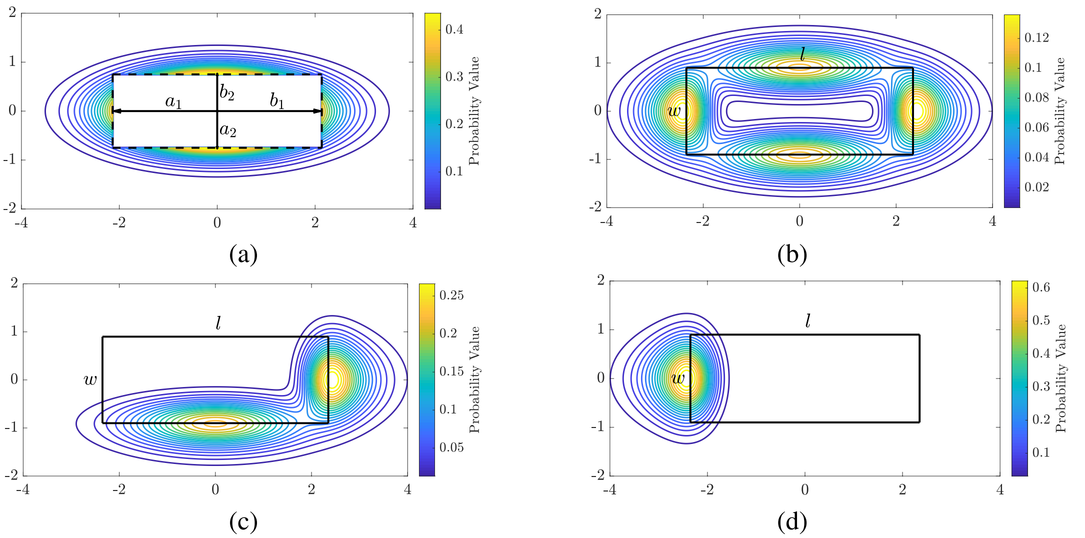
For automotive radar sensors, the measurements may appear on sides of the object that are not in sight of the sensors due to multi-path propagation. If the multi-path measurements only make up a tiny fraction of the entire training data as in the nuScenes case, fully data-driven models may present degraded estimation performance when the online measurements are sparse. This effect can be observed from relatively large standard deviations of all estimation errors for the GMM-PMHT and B-spline model in Table I. Particularly, the large standard deviations were contributed by those trajectory estimates produced by the GMM-PMHT and B-spline model that sometimes diverge from the ground truth. On the other hand, by having fewer model parameters in the learned model, the HTG implementations presented good robustness against the multi-path measurements, which is a crucial factor in automotive applications.

VII. Conclusion and Future Work

In this paper, we have proposed a new measurement model for extended object tracking with automotive radar by modeling the measurement spatial density as a HTG, which can be learned from real-world automotive radar data. We have also presented two modified random matrix update methods for the learned HTG model based on two different density

[6] Same as the model training described in [23], the variational estimation model from the sklearn Gaussian mixture library [37] was used to learn the GMMs, in which 20 components were chosen for the learning process.

[7] All the implementations are in MATLAB, run on a laptop with 2.7-GHz Intel Core i5 processor.



(a) (b) (c) (d)

Fig. 9. Probability density functions of (a) a truncated Gaussian centered at the origin with $l = 4.7$, $w = 1.8$, $a_1 = b_1 = 2.14$, $a_2 = b_2 = 0.75$; (b) a HTG with $R = 0.09\mathrm{I}_2$ in the object centered frame. Both horizontal and vertical axes are in the unit of meters; (c) a HTG with $a_2 = 0.75$, $b_1 = 2.14$ and $a_1 = b_2 = +\infty$; (d) a HTG with $a_1 = 2.14$, $a_1 = b_1 = b_2 = +\infty$.

representations. The effectiveness of the proposed approaches have been validated on both synthetic and experimental data.

To apply the proposed methods in more practical applications, follow-up directions include the integration of range rate measurements, see, e.g., [12] and the extension to multiple extended object tracking, see, e.g., [32], [38]. It would also be interesting to generalize the measurement updates for HTG models to other random matrix based update equations for Gaussian spatial densities [12], [32], [38]–[40].

Appendix A

In this appendix, we show examples of pdfs of truncated Gaussian and HTG introduced in [24], [25].

The truncated Gaussian distribution illustrated in Fig. 9 (a) is a truncated version of $\mathcal{N}(0_{2\times1}, \mathrm{diag}([l^2/4, w^2/4]))$. The HTG distributions illustrated in the other three plots are all based on the truncated Gaussian in Fig. 9 (a) but with different truncation bounds.

Appendix B

In this appendix, we present the pseudo code for sampling the HTG distribution illustrated in Fig. 2.

Algorithm 3 Sampling a HTG distribution

```
Input: HTG parameters $a_1, b_1, a_2, b_2, r_1, r_2, \rho, \theta$.
Output: Sample $u$.
$y \sim \mathcal{N}(0_{2\times1}, \rho\mathrm{I}_2)$.
while $a_1 < y_x < b_1$ and $a_2 < y_y < b_2$ do
    $y \sim \mathcal{N}(0_{2\times1}, \rho\mathrm{I}_2)$.
end while
$u \sim \mathcal{N}(y, \mathrm{diag}([r_1, r_2]))$.
$u = M(\theta)u$.
```

Appendix C

In this appendix, we derive (12).

We start with the univariate case. Suppose that $\tilde{u} \sim \mathcal{N}(y, r)$ a Gaussian distribution with mean y and variance r. Also, suppose that $y \sim \mathcal{TN}(0, \rho, [-a, b])$ a truncation of Gaussian distribution with mean 0 and variance ρ on the interval $[-a, b]$. The pdf of y is

$$p(y) = \frac{\mathcal{N}(y; 0, \rho)\mathbf{1}_{[-a,b]}(y)}{\int_{-a}^{b}\mathcal{N}(y; 0, \rho)} = \frac{\mathcal{N}(y; 0, \rho)\mathbf{1}_{[-a,b]}(y)}{\Phi(b\rho^{-1/2}) - \Phi(-a\rho^{-1/2})}. \tag{41}$$

As $p(\tilde{u}|y)p(y) = p(y|\tilde{u})p(\tilde{u})$, we have that

$$\begin{aligned}&\frac{\mathcal{N}(\tilde{u}; y, r)\mathcal{N}(y; 0, \rho)\mathbf{1}_{[-a,b]}(y)}{\Phi(b\rho^{-1/2}) - \Phi(-a\rho^{-1/2})} \\ &\qquad = \frac{\mathcal{N}\left(y; \frac{\tilde{u}\rho}{r+\rho}, \frac{r\rho}{r+\rho}\right)\mathcal{N}(\tilde{u}; 0, r+\rho)\mathbf{1}_{[-a,b]}(y)}{\Phi(b\rho^{-1/2}) - \Phi(-a\rho^{-1/2})}.\end{aligned} \tag{42}$$

Using (42), the pdf of hierarchical truncated Gaussian variable $\tilde{u}$ can be written as

$$\begin{aligned}p(\tilde{u}) &= \int p(\tilde{u}|y)p(y)dy \qquad (43)\\ &= \frac{\mathcal{N}(\tilde{u}; 0, r+\rho)\int_{-a}^{b}\mathcal{N}\left(y; \frac{\tilde{u}\rho}{r+\rho}, \frac{r\rho}{r+\rho}\right)dy}{\Phi(b\rho^{-1/2}) - \Phi(-a\rho^{-1/2})} \\ &= \frac{\mathcal{N}(\tilde{u}; 0, r+\rho)\left(\Phi\left(\frac{b - \frac{\tilde{u}\rho}{r+\rho}}{\sqrt{\frac{r\rho}{r+\rho}}}\right) - \Phi\left(\frac{-a - \frac{\tilde{u}\rho}{r+\rho}}{\sqrt{\frac{r\rho}{r+\rho}}}\right)\right)}{\Phi(b\rho^{-1/2}) - \Phi(-a\rho^{-1/2})}.\end{aligned}$$

We proceed to consider the bivariate case. Suppose that $u \sim \mathcal{N}(y, M(\theta)\,\mathrm{diag}([r_1, r_2])M(\theta)^{\mathrm{T}})$ and $y \sim \mathcal{TN}(0_{2\times1}, \rho\mathrm{I}_2)$ a truncated Gaussian distribution with density support D specified by $B = (a_1, a_2, b_1, b_2)$ and θ as illustrated in Fig. 2 (a). We also write $\tilde{u} = M(-\theta)u$. The pdf of hierarchical truncated Gaussian variable u is

$$\begin{aligned}p(u) &= \int p(u|y)p(y)dy \qquad (44)\\ &= \frac{1}{c_D}\left(\int \mathcal{N}\left(\tilde{u}; y, \mathrm{diag}\left([r_1, r_2]\right)\right)\mathcal{N}\left(y; 0_{2\times1}, \rho\mathrm{I}_2\right)dy\right.\end{aligned}$$

$$
\begin{aligned}
&- \int_{\mathbb{R}^2\setminus D} \mathcal{N}\left(\tilde{u}; y, \operatorname{diag}\left([r_1, r_2]\right)\right) \mathcal{N}\left(y; 0_{2\times 1}, \rho \mathrm{I}_2\right) dy \Bigg) \\
&= \frac{1}{c_D} \Bigg(\mathcal{N}(\tilde{u}_x; 0, r_1+\rho) \mathcal{N}(\tilde{u}_y; 0, r_2+\rho) \\
&\quad - \mathcal{N}(\tilde{u}_x; 0, r_1+\rho) \int_{-a_1}^{b_1} \mathcal{N}\left(y; \frac{\tilde{u}_x \rho}{r_1+\rho}, \frac{r_1 \rho}{r_1+\rho}\right) dy \\
&\quad \times \mathcal{N}(\tilde{u}_y; 0, r_2+\rho) \int_{-a_2}^{b_2} \mathcal{N}\left(y; \frac{\tilde{u}_y \rho}{r_2+\rho}, \frac{r_2 \rho}{r_2+\rho}\right) dy \Bigg)
\end{aligned}
$$

where

$$
\begin{aligned}
c_D &= 1 - c_{\mathbb{R}^2\setminus D} \\
&= 1 - \big((\Phi(b_1 \rho^{-1/2}) - \Phi(-a_1 \rho^{-1/2})) \\
&\qquad \times (\Phi(b_2 \rho^{-1/2}) - \Phi(-a_2 \rho^{-1/2})) \big).
\end{aligned} \tag{45}
$$

We then take the sum of negative logarithm of $p(u^j)$ for $j = 1, \ldots, N$, which finally yields the objective function in (12).

Appendix D

In this appendix, we show how to use coordinate descent optimization to solve (12).

Let the multivariate function to be minimized in (12) be $f(\mathbf{x})$ where $\mathbf{x} = (\rho, \theta, r_1, r_2, a_1, b_1, a_2, b_2)$, let the ith variable in $\mathbf{x}$ be $\mathbf{x}_i$, let the value of $\mathbf{x}$ at the tth iteration be $\mathbf{x}^t$ and let the constrained space on the ith variable be $\mathbb{C}_i$. We start with some initial values $\mathbf{x}^0$. For the tth iteration, we pick frame i from $i = 1, 2, \ldots, 8$ and solve the constrained univariate optimization problem

$$
x_i^{t+1} = \underset{x_i \in \mathbb{C}_i}{\arg\min}\, f(x_1^t, \ldots, x_{i-1}^t, x_i, x_{i+1}^t, \ldots, x_8^t). \tag{46}
$$

If at some point, $f(\mathbf{x}^t) - f(\mathbf{x}^{t+1})$ is smaller than a threshold at every frame direction, then we assume that the optimum is reached.

As the objective function is not strictly smooth and convex, the algorithm does not necessarily converge to a global optimum. Instead, we use an empirical study to show that the ML estimate obtained is precise with moderate amount of training data. Suppose that the training data consists of 10000 data points randomly generated using the HTG model illustrated in Fig. 2. The objective is to see how close the ML estimates of the model parameters obtained via coordinate descent optimization are to the ground truth.

In the simulation, each constrained optimization subproblem is solved using `fmincon` in `MATLAB` (with default options), the convergence threshold is set to 10^{-6} and the initial values are $\mathbf{x}_0 = (0.25, 0, 0.04, 0.04, 0.5, 0.5, 0.5, 0.5)$. The estimation errors obtained over 100 Monte Carlo runs are summarized in Table II. The results show that the ML estimate of HTG model parameters obtained via coordinate descent optimization can be considered precise enough.

TABLE II
Mean/Maximum ML Estimation Absolute Errors

	ρ	θ	r_1	r_2	a_1	b_1	a_2	b_2
Mean	0.004	0.012	0.005	0.004	0.008	0.014	0.005	0.010
Max	0.019	0.034	0.022	0.018	0.043	0.069	0.015	0.057

Appendix E

In this appendix, we compute the mean and covariance of $\mathcal{TN}(y_k^c; 0_{2\times 1}, \rho \mathrm{I}_2, D_k^c)$ where $D_k^c = \mathbb{R}^2 \setminus D_k$. An illustration of the pdf of $\mathcal{TN}(y_k; 0_{2\times 1}, \rho \mathrm{I}_2, D_k)$ is given in Fig. 2 (a).

We start with transforming truncated Gaussian random variable y_k^c to a new frame by $\tilde{y}_k^c = M(-\theta) y_k^c$, such that the orientation of the truncation area is now aligned with the axes. After the transformation, the computations of the mean and covariance of $\tilde{y}_k^c$ can be decomposed into the computations of the mean and variance of two univariate truncated Gaussians. Specifically, we have that

$$
\tilde{y}_{k,x}^c \sim \mathcal{TN}(0, \rho, [-a_1, b_1]), \tag{47a}
$$

$$
\tilde{y}_{k,y}^c \sim \mathcal{TN}(0, \rho, [-a_2, b_2]). \tag{47b}
$$

Let $\alpha_i = -a_i \rho^{-1/2}$ and $\beta_i = b_i \rho^{-1/2}$ for $i \in \{1, 2\}$. Also, let $\varphi(\cdot)$ denote the pdf of the standard normal distribution. The mean and variance of $\tilde{y}_{k,x}^c$ and $\tilde{y}_{k,y}^c$ are [41, Sec. 10.1]

$$
\mathbb{E}\left[\tilde{y}_{k,x}^c\right] = \rho^{1/2} \frac{\varphi(\alpha_1) - \varphi(\beta_1)}{\Phi(\beta_1) - \Phi(\alpha_1)}, \tag{48a}
$$

$$
\mathbb{E}\left[\tilde{y}_{k,y}^c\right] = \rho^{1/2} \frac{\varphi(\alpha_2) - \varphi(\beta_2)}{\Phi(\beta_2) - \Phi(\alpha_2)}, \tag{48b}
$$

$$
\begin{aligned}
\operatorname{Var}\left[\tilde{y}_{k,x}^c\right] = \rho \Bigg(& 1 + \frac{\alpha_1 \varphi(\alpha_1) - \beta_1 \varphi(\beta_1)}{\Phi(\beta_1) - \Phi(\alpha_1)} \\
& - \left(\frac{\varphi(\alpha_1) - \varphi(\beta_1)}{\Phi(\beta_1) - \Phi(\alpha_1)} \right)^2 \Bigg),
\end{aligned} \tag{48c}
$$

$$
\begin{aligned}
\operatorname{Var}\left[\tilde{y}_{k,y}^c\right] = \rho \Bigg(& 1 + \frac{\alpha_2 \varphi(\alpha_2) - \beta_2 \varphi(\beta_2)}{\Phi(\beta_2) - \Phi(\alpha_2)} \\
& - \left(\frac{\varphi(\alpha_2) - \varphi(\beta_2)}{\Phi(\beta_2) - \Phi(\alpha_2)} \right)^2 \Bigg).
\end{aligned} \tag{48d}
$$

After compensating the frame transformation, the mean and covariance of y_k^c can be written as

$$
\begin{aligned}
\mathbb{E}\left[y_k^c\right] &= M(\theta) \left[\mathbb{E}\left[\tilde{y}_{k,x}^c\right], \mathbb{E}\left[\tilde{y}_{k,y}^c\right] \right]^{\mathrm{T}}, \\
\operatorname{Cov}\left[y_k^c\right] &= M(\theta) \operatorname{diag}\left(\left[\operatorname{Var}\left[\tilde{y}_{k,x}^c\right], \operatorname{Var}\left[\tilde{y}_{k,y}^c\right] \right] \right) M(\theta)^{\mathrm{T}}.
\end{aligned} \tag{49}
$$

References

[1] S. Sun, A. P. Petropulu, and H. V. Poor, "MIMO radar for advanced driver-assistance systems and autonomous driving: Advantages and challenges," *IEEE Signal Processing Magazine*, vol. 37, no. 4, pp. 98–117, 2020.

[2] K. Granström, M. Baum, and S. Reuter, "Extended object tracking: Introduction, overview, and applications," *Journal of Advances in Information Fusion*, vol. 12, no. 2, 2017.

[3] P. Broßeit, M. Rapp, N. Appenrodt, and J. Dickmann, "Probabilistic rectangular-shape estimation for extended object tracking," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 279–285.

[4] X. Cao, J. Lan, X. R. Li, and Y. Liu, "Extended object tracking using automotive radar," in *2018 21st International Conference on Information Fusion (FUSION)*. IEEE, 2018, pp. 1–5.

[5] M. Baum and U. D. Hanebeck, "Extended object tracking with random hypersurface models," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 50, no. 1, pp. 149–159, 2014.

[6] N. Wahlström and E. Özkan, "Extended target tracking using Gaussian processes," *IEEE Trans. on Signal Processing*, vol. 63, no. 16, pp. 4165–4178, 2015.

[7] K. Thormann, M. Baum, and J. Honer, "Extended target tracking using Gaussian processes with high-resolution automotive radar," in *2018 21st International Conference on Information Fusion (FUSION)*. IEEE, 2018, pp. 1764–1770.

[8] H. Kaulbersch, J. Honer, and M. Baum, "A Cartesian B-spline vehicle model for extended object tracking," in *2018 21st International Conference on Information Fusion (FUSION)*, 2018, pp. 1–5.

[9] J. W. Koch, "Bayesian approach to extended object and cluster tracking using random matrices," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 44, no. 3, pp. 1042–1059, 2008.

[10] 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.

[11] U. Orguner, "A variational measurement update for extended target tracking with random matrices," *IEEE Trans. on Signal Processing*, vol. 60, no. 7, pp. 3827–3834, 2012.

[12] M. Schuster, J. Reuter, and G. Wanielik, "Probabilistic data association for tracking extended targets under clutter using random matrices," in *2015 Fusion*, 2015, pp. 961–968.

[13] S. Yang and M. Baum, "Tracking the orientation and axes lengths of an elliptical extended object," *IEEE Trans. on Signal Processing*, vol. 67, no. 18, pp. 4720–4729, 2019.

[14] N. J. Bartlett, C. Renton, and A. G. Wills, "A closed-form prediction update for extended target tracking using random matrices," *IEEE Trans. on Signal Processing*, vol. 68, pp. 2404–2418, 2020.

[15] P. Berthold, M. Michaelis, T. Luettel, D. Meissner, and H.-J. Wuensche, "Radar reflection characteristics of vehicles for contour and feature estimation," in *2017 SDF*. IEEE, 2017, pp. 1–6.

[16] ——, "An abstracted radar measurement model for extended object tracking," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 3866–3872.

[17] L. Hammarstrand, L. Svensson, F. Sandblom, and J. Sorstedt, "Extended object tracking using a radar resolution model," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2371–2386, 2012.

[18] L. Hammarstrand, M. Lundgren, and L. Svensson, "Adaptive radar sensor model for tracking structured extended objects," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 1975–1995, 2012.

[19] C. Knill, A. Scheel, and K. Dietmayer, "A direct scattering model for tracking vehicles with high-resolution radars," in *2016 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2016, pp. 298–303.

[20] 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," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 621–11 631.

[21] P. Broßeit, B. Duraisamy, and J. Dickmann, "The volcanormal density for radar-based extended target tracking," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 1–6.

[22] A. Scheel and K. Dietmayer, "Tracking multiple vehicles using a variational radar model," *IEEE Trans. on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3721–3736, 2018.

[23] H. Kaulbersch, J. Honer, and M. Baum, "EM-based extended target tracking with automotive radar using learned spatial distribution models," in *2019 22th International Conference on Information Fusion (FUSION)*, 2019, pp. 1–8.

[24] Y. Xia, P. Wang, K. Berntorp, T. Koike-Akino, H. Mansour, M. Pajovic, P. Boufounos, and P. V. Orlik, "Extended object tracking using hierarchical truncation measurement model with automotive radar," in *ICASSP 2020*, 2020, pp. 4900–4904.

[25] Y. Xia, P. Wang, K. Berntorp, H. Mansour, P. Boufounos, and P. V. Orlik, "Extended object tracking using hierarchical truncation model with partial-view measurements," in *2020 IEEE SAM*, 2020, pp. 1–5.

[26] Y. Xia, P. Wang, K. Berntorp, P. Boufounos, P. V. Orlik, K. Granström, and L. Svensson, "Extended object tracking with automotive radar using learned structural measurement model," in *2020 IEEE Radar Conference (RadarConf)*. IEEE, 2020, pp. 1–6.

[27] G. Vivone, K. Granström, P. Braca, and P. Willett, "Multiple sensor measurement updates for the extended target tracking random matrix model," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 53, no. 5, pp. 2544–2558, 2017.

[28] S. Bordonaro, P. Willett, and Y. Bar-Shalom, "Consistent linear tracker with converted range, bearing, and range rate measurements," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 53, no. 6, pp. 3135–3149, 2017.

[29] K. Thormann and M. Baum, "Optimal fusion of elliptic extended target estimates based on the Wasserstein distance," in *FUSION*, 2019, pp. 1–6.

[30] A. Gupta and D. Nagar, "Monographs and surveys in pure and applied mathematics," in *Matrix Variate Distributions*. Chapman and Hall, 2000.

[31] G. Vivone, P. Braca, K. Granström, and P. Willett, "Multistatic Bayesian extended target tracking," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 52, no. 6, pp. 2626–2643, 2016.

[32] K. Granström and U. Orguner, "A PHD filter for tracking multiple extended targets using random matrices," *IEEE Trans. on Signal Processing*, vol. 60, no. 11, pp. 5657–5671, 2012.

[33] J. S. Liu and R. Chen, "Blind deconvolution via sequential imputations," *Journal of the American Statistical Association*, vol. 90, no. 430, pp. 567–576, 1995.

[34] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.

[35] S. Yang, M. Baum, and K. Granström, "Metrics for performance evaluation of elliptic extended object tracking methods," in *2016 IEEE MFI*, 2016, pp. 523–528.

[36] C. R. Givens, R. M. Shortt *et al.*, "A class of Wasserstein metrics for probability distributions." *The Michigan Mathematical Journal*, vol. 31, no. 2, pp. 231–240, 1984.

[37] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, "Scikit-learn: Machine learning in Python," *the Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[38] K. Granström, M. Fatemi, and L. Svensson, "Poisson multi-Bernoulli mixture conjugate prior for multiple extended target filtering," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 56, pp. 208–225, 2019.

[39] T. Ardeshiri, U. Orguner, and F. Gustafsson, "Bayesian inference via approximation of log-likelihood for priors in exponential family," in *CoRR*, 2015, pp. 1–13.

[40] B. Liu, R. Tharmarasa, R. Jassemi, D. Brown, and T. Kirubarajan, "Extended target tracking with multipath detections, terrain-constrained motion model and clutter," *IEEE Trans. on Intelligent Transportation Systems*, 2020.

[41] N. L. Johnson, S. Kotz, and N. Balakrishnan, *Continuous univariate distributions*. John Wiley & Sons, Ltd, 1995.

Yuxuan Xia was born in Wuhan, China, 1993. He received the B.Sc. degree in Engineering of Internet of Things from the Jiangnan University, Wuxi, China in 2015 and the M.Sc. degree in Communication Engineering from Chalmers University of Technology, Gothenburg, Sweden in 2017. He is currently pursuing his Ph.D. degree in the Department of Electrical and Engineering, Chalmers University of Technology. His main research focus is on object tracking.

Pu Wang (S'05-M'12-SM'18) received the Ph.D. degree in Electrical Engineering from the Stevens Institute of Technology, Hoboken, NJ, USA, in 2011.

He is now a Principal Research Scientist at the Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, where he was an intern in the summer of 2010. His current research interests include signal processing, Bayesian inference, machine learning, and their applications to (mmWave and THz) sensing, wireless communications, networks, and autonomous driving.

Dr. Wang received the IEEE Jack Neubauer Memorial Award from the IEEE Vehicular Technology Society in 2013, the Outstanding Paper Award from the IEEE AFICON Conference in 2011, the Outstanding Doctoral Thesis in Electrical Engineering Award in 2011, the Edward Peskin Award in 2011, the Francis T. Boesch Award in 2008, and the Outstanding Research Assistant Award in 2007 from the Stevens Institute of Technology. He is an Associate Editor for *IEEE Signal Processing Letters*.

Karl Berntorp received the M.Sc. degree in engineering physics and the Ph.D. degree in automatic control from Lund University, Lund, Sweden, in 2009 and 2014, respectively.

In 2008, he was a Visiting Researcher with Daimler AG, Sindelfingen, Germany. Since 2014, he has been with Mitsubishi Electric Research Laboratories, Cambridge, MA, USA. His work includes the design and implementation of estimation, constrained control, motion planning, and learning algorithms. He is the author of more than 75 papers in journals and conferences and has more than ten granted patents. His research interests include statistical signal processing, Bayesian inference, sensor fusion, and optimization-based control, with applications to automotive, transportation, navigation, and communication systems. Dr. Berntorp is senior member of IEEE and associated editor of the IEEE Technology Conferences Editorial Board.

Lennart Svensson was born in Älvängen, Sweden in 1976. He received the M.S. degree in electrical engineering in 1999 and the Ph.D. degree in 2004, both from the Chalmers University of Technology, Gothenburg, Sweden. He is currently a Professor of Signal Processing with the Chalmers University of Technology. His main research interests include machine learning and Bayesian inference in general, and nonlinear filtering, deep learning, and tracking in particular. He has organized a massive open online course on multiple object tracking, available on edX and YouTube, and received paper awards at the International Conference on Information Fusion in 2009, 2010, 2017, and 2019.

Karl Granström (Member, IEEE) received the M.Sc. degree in applied physics and electrical engineering and the Ph.D. degree in automatic control from Linköping University, Sweden, in May 2008 and November 2012, respectively. He is currently a Perception Specialist at Embark Trucks Inc., San Francisco, CA, USA. Before joining Embark, he worked at the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden, first as a Postdoctoral Researcher and then as a Researcher. He held postdoctoral positions at the Department of Electrical and Computer Engineering, University of Connecticut, USA, from September 2014 to August 2015, and at the Department of Electrical Engineering, Linköping University, from December 2012 to August 2014. His research interests include estimation theory, multiple model estimation, sensor fusion, and target tracking, especially for extended targets. He received paper awards at the Fusion 2011 and Fusion 2012 conferences. He has organized several workshops and tutorials on the topic of Multiple Extended Target Tracking and Sensor Fusion. At Fusion 2018, the International Society for Information Fusion (ISIF) awarded him the ISIF Young Investigator Award for his contributions to extended target tracking

Hassan Mansour (S’99-M’09-SM’17) received the B.E. degree in computer and communications engineering from the American University of Beirut, Beirut, Lebanon, in 2003, and the M.A.Sc. degree in electrical and computer engineering and the Ph.D. degree in electrical and computer engineering from The University of British Columbia, Vancouver, BC, Canada, in 2005 and 2009, respectively. Between January 2010 and January 2013, he was a Postdoctoral Research Fellow with the Department of Computer Science, the Mathematics Department, and the Department of Earth, Ocean, and Atmospheric Sciences, The University of British Columbia. He is currently a Senior Principal Research Scientist with Mitsubishi Electric Research Laboratories, Cambridge, MA, USA. His research interests are in inverse problems, compressed sensing, sparse signal reconstruction, image enhancement, and scalable video compression and transmission. His current research is focused on the design of efficient acquisition schemes and reconstruction algorithms for natural images, radar sensing, video analytics, and inverse scattering problems. Dr. Mansour is a member of the IEEE Computational Imaging Technical Committee and the IEEE Sensor Array and Multichannel Technical Committee. He is also an Associate Editor for the IEEE Transactions on Signal Processing.

Petros T. Boufounos (S’02-M’06-SM’13) is a Senior Principal Research Scientist and the Computational Sensing Team Leader at Mitsubishi Electric Research Laboratories (MERL). Dr. Boufounos completed his undergraduate and graduate studies at MIT. He received the S.B. degree in Economics in 2000, the S.B. and M.Eng. degrees in Electrical Engineering and Computer Science (EECS) in 2002, and the Sc.D. degree in EECS in 2006. Between September 2006 and December 2008, he was a postdoctoral associate with the Digital Signal Processing Group at Rice University. Dr. Boufounos joined MERL in January 2009, where he has been heading the Computational Sensing Team since 2016.

Dr. Boufounos’ immediate research focus includes signal acquisition and processing, computational sensing, inverse problems, frame theory, quantization, and data representations. He is also interested in how signal acquisition interacts with other fields that use sensing extensively, such as machine learning, robotics, and dynamical system theory. Dr. Boufounos has served as an Area Editor and a Senior Area Editor for the IEEE signal processing letters. He has been a part of the SigPort editorial board and is currently a member of the IEEE Signal Processing Society Theory and Methods technical committee and and an Associate Editor at IEEE Transactions on Computational Imaging. He was also named IEEE SPS Distinguished Lecturer for 2019-2020.

Philip V. Orlik (M’97-SM’12) received the B.E. degree in 1994 and the M.S. degree in 1997 both from the State University of New York at Stony Brook. In 1999 he earned his Ph. D. in electrical engineering also from SUNY Stony Brook. From 1999 to 2000 he was a Simulation and Modelling Engineer for the Mitre Corporation. In 2000 he joined Mitsubishi Electric Research Laboratories, (MERL) Inc. located in Cambridge, MA as a Member of Technical Staff. He is he is currently the Manager of the Signal Processing Group. His primary research focus is on advanced signal processing methods for various applications including radar, material sensing, advanced wireless and wired communications, sensor and IoT networks. Other research interests include vehicular/car-to-car communications, mobility modeling, performance analysis, and queuing theory.