3D-Extended Object Tracking and Shape Classification with a Lidar Sensor using Random Matrices and Virtual Measurement Models

Patrick Hoher*, Tim Baur*, Johannes Reuter*, Dennis Griesser†
University of Applied Sciences Konstanz
*Institute of System Dynamics
†Institute for Optical Systems
Konstanz, Germany
{phoher,tbaur,jreuter,dgriesser}@htwg-konstanz.de

Felix Govaers, Wolfgang Koch
Fraunhofer FKIE
Dept. Sensor data and Information Fusion
Wachtberg, Germany
{felix.govaers,wolfgang.koch}@fkie.fraunhofer.de

Abstract—**In extended object tracking, random matrices are commonly used to filter the mean and covariance matrix from measurement data. However, the relation from mean and covariance matrix to the extension parameters can become challenging when a lidar sensor is used. To address this, we propose virtual measurement models to estimate those parameters iteratively by adapting them, until the statistical moments of the measurements they would cause, match the random matrix result. While previous work has focused on 2D shapes, this paper extends the methodology to encompass 3D shapes such as cones, ellipsoids and rectangular cuboids. Additionally, we introduce a classification method based on Chamfer distances for identifying the best-fitting shape when the object's shape is unknown. Our approach is evaluated through simulation studies and with real lidar data from maritime scenarios. The results indicate that a cone is the best representation for sailing boats, while ellipsoids are optimal for motorboats.**

Index Terms—**Extended object tracking, random matrices, lidar, extension estimation, virtual measurement model, ellipsoid, cone, cuboid, Chamfer distance**

I. INTRODUCTION

The goal of extended object tracking (EOT) is to determine the kinematic state and the extent parameters of an object from measurement data. In some applications, the shape is known or assumed to be, for example, elliptical [1]–[11], rectangular [9], [12]–[14] or star-convex [11], [15]. In 3D-EOT, commonly used shapes include ellipsoids [1]–[4], cones [16], cylinders [17] or rectangular cuboids [18].

In addition to state and extension estimation, classification plays a significant role in tracking applications. Joint EOT and classification problems may involve various classes such as cars [19], pedestrians [19], animals [20] and ships [21] or focus on specific shapes [22][23] if the shape is unknown. One of the most commonly used methods for EOT is the random matrix (RM) approach [2]. Although the original paper [1] was published in 2008, approaches based on random matrix are still state of the art today [24], [25] for several reasons: It is robust against measurement noise, easy to implement and computational efficient. Nevertheless, a fundamental assumption of this method, i.e., the normal or uniform distribution of measurements over the object's extent, is challenged when employing lidar measurement models based on ray tracing [5]. In such models, measurements are asymmetrically distributed along the object's contour, affecting the applicability of RMs. To combine the robustness of RM with the precision of ray tracing, an adaptation with virtual measurement models (VMMs) [5]–[7], [23] has been proposed. This adaptation involves varying a given estimate, generating artificial measurements using the VMM and comparing their statistical moments to the RM result until they match [5]. In scenarios where the shape is unknown, multiple VMMs corresponding to various shapes can be used and the comparison of shape-specific artificial measurements with real measurements can be utilized for shape classification [23].

While VMMs have primarily been employed in 2D scenarios, this work extends their applicability to 3D shapes, enhances classification using Chamfer distances [26]–[28], and provides comprehensive results, including analyses of real lidar data from maritime scenarios.

II. BACKGROUND

A. Random matrices

The random matrix (RM) approach [2] is a recursive algorithm used to estimate the normal distributed kinematic state density and the inverse Wishart distributed extension density of an object. Random matrices are applicable to both 2D and 3D object tracking. Detailed prediction and update equations can be found in [2]. In general, random matrices can be used to filter the center of gravity (CoG) and the covariance matrix of measurements, independently of the object's shape and measurement model [5]. But without adaptation, the relation to position and extension parameters remains unclear when the measurements are not normally or uniformly distributed.

B. Virtual measurement model (VMM) framework

The EOT framework with VMMs [5] that is also used within this paper, is shown in Fig. 1. The input $\underline{u}_k = (x, y, z, l, w, h)^\top$ includes all state elements and parameters

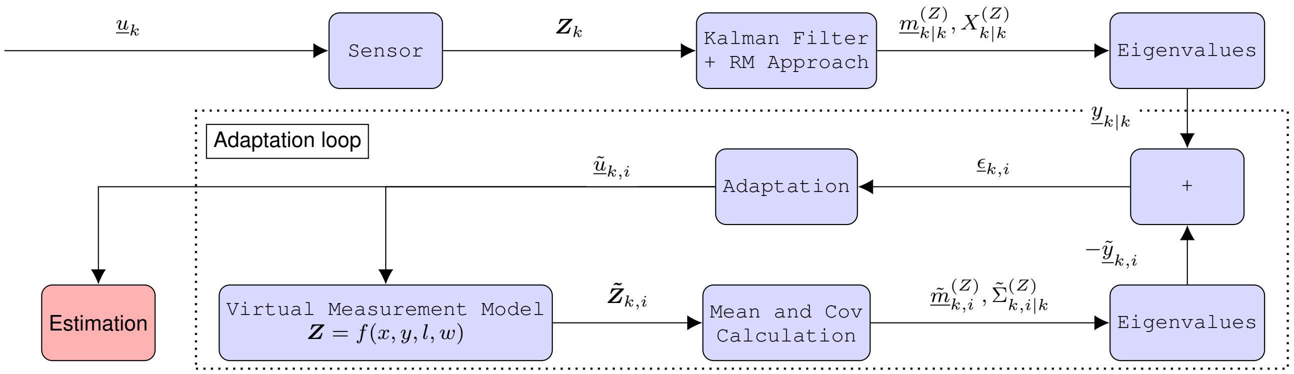


Fig. 1: EOT framework with RM and an adaptation algorithm using the VMM. Figure from [5, p. 231][23, p. 2].

for adjustment, such as 3D position (x, y, z), length l, width w, and height h [5, p. 232]. Velocities and orientation angles are assumed to be adequately estimated using the Kalman filter with RM [2] and hence, do not require additional compensation. First, the Kalman filter with random matrices from [2] is applied to the measurements $\boldsymbol{Z}_k$, without incorporating the lidar measurement model. The result $\underline{y}_k = \left(z_x, z_y, z_z, \sqrt{e_l}, \sqrt{e_w}, \sqrt{e_h}\right)$ contains the filtered CoG (z_x, z_y, z_z) and the square roots of the eigenvalues (e_l, e_w, e_h) of the filtered covariance matrix. Each eigenvalue corresponds to an extension parameter identified by its indices. Prior work in 2D-EOT [5]–[7], [23] assumed that e_l represents the largest eigenvalue and e_w represents the smallest. This restriction will be lifted by introducing an eigenvector-based assignment.

The adaptation process utilizing the VMM begins by generating virtual measurements $\boldsymbol{\tilde{Z}}_{k,i}$ for the current estimate $\underline{\tilde{u}}_{k,i}$. Subsequently, their mean $\underline{\tilde{m}}^{(Z)}_{k,i}$ and covariance $\tilde{\Sigma}^{(Z)}_{k,i|k}$ are computed. The mean and eigenvalues of the covariance matrix are stored in the vector $\underline{\tilde{y}}_{k,i} = \left(\tilde{z}_x, \tilde{z}_y, \tilde{z}_z, \sqrt{\tilde{e}_l}, \sqrt{\tilde{e}_w}, \sqrt{\tilde{e}_h}\right)^\top$, mirroring the structure of $\underline{y}_k$. The VMM's output $\underline{\tilde{y}}_{k,i}$ is then compared to the RM result $\underline{y}_k$, and the resulting error $\epsilon_{k,i}$ is used to update the estimate according to [23, p. 2]: $\underline{\tilde{u}}_{k,i+1} = \underline{\tilde{u}}_{k,i} + K_i \underline{\epsilon}_{k,i}$.

In each time step k, the adaptation loop is executed several times (index i) to minimize ϵ_k. Then, the outputs $\underline{y}_k$ and $\underline{\tilde{y}}_{k,i}$ match, and, assuming full observability, the inputs must also match and $\underline{\tilde{u}}_{k,i}$ is considered as the position and extent estimation. Additional background, including VMM input initialization and VMM input prediction, is provided in [5].

C. Chamfer distances

The Chamfer distance [26] is a metric used to calculate the distance between two point clouds by measuring the shortest distance from each point in one shape to the nearest point in the other shape. The Chamfer distance between two 3D point clouds S_1 and S_2 is given by [27, p. 4] [28, p. 4]:

$$d_C(S_1, S_2) = \frac{1}{|S_1|} \sum_{x \in S_1} \min_{j \in S_2} ||x - y||^2 + \frac{1}{|S_2|} \sum_{y \in S_2} \min_{x \in S_1} ||y - x||^2, \tag{1}$$

whereas $|S_i|$ is the number of elements in S_i.

III. 3D MEASUREMENT MODELS

The first contributions of this paper is a ray tracing method for 3D-shapes, including ellipsoids, cones and rectangular cuboids. It is used to generate artificial lidar measurements that are required for the VMM adaptation. For the following calculations, we distinguish between a local coordinate system ENU (xEast, yNorth, zUp) and a body fixed coordinate system BODY. BODY can also be denoted as target coordinate system [5, p. 231]. A detailed explanation about coordinate systems and their transformations is given by Fossen in [29].

A. Ellipsoid

The surface of a triaxial ellipsoid is defined by:

$$\frac{x^2}{l^2} + \frac{y^2}{w^2} + \frac{z^2}{h^2} = \frac{1}{4}, \tag{2}$$

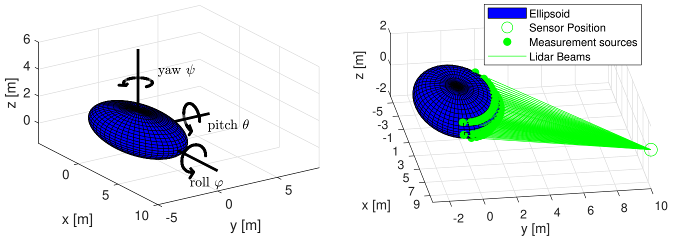
with length l, width w and height h for the special case, that the ellipsoid is unrotated and positioned at the origin. In spherical coordinates, its surface can be described using

$$x_1 = \frac{l}{2} \sin(t_2) cos(t_1), \tag{3}$$

$$y_1 = \frac{w}{2} \sin(t_2) sin(t_2), \tag{4}$$

$$z_1 = \frac{h}{2} \cos(t_2), \tag{5}$$

with azimuth angle t_1 and inclination t_2. In general, however, the ellipsoid is located at (x_0, y_0, z_0) and rotated by pitch θ, yaw ψ and roll φ (see Fig. 2a). This transformation from ENU



(a) Pitch θ, yaw ψ and roll φ

(b) Ray tracing

Fig. 2: 3D Ellipsoid

to BODY coordinates is expressed by [29, p. 24, adjusted for zUp]:

$$\begin{pmatrix} x\\ y\\ z\end{pmatrix} = \begin{pmatrix} x_0\\ y_0\\ z_0\end{pmatrix} + \underbrace{\begin{pmatrix}1&0&0\\0&\cos(\varphi)&-\sin(\varphi)\\0&\sin(\varphi)&\cos(\varphi)\end{pmatrix}}_{R_\varphi \text{ for roll}} \cdot \underbrace{\begin{pmatrix}\cos(\theta)&0&-\sin(\theta)\\0&1&0\\\sin(\theta)&0&\cos(\theta)\end{pmatrix}}_{R_\theta \text{ for pitch}} \underbrace{\begin{pmatrix}\cos(\psi)&-\sin(\psi)&0\\\sin(\psi)&\cos(\psi)&0\\0&0&1\end{pmatrix}}_{R_\psi \text{ for yaw}} \begin{pmatrix}x_1\\y_1\\z_1\end{pmatrix}. \tag{6}$$

Since the ellipsoidal equation in (2) is less complex than in ENU (6), the following intersection calculation for ray tracing is performed in BODY coordinates. Note that in [5, p. 231], the same procedures were used for 2D ellipses. Fig. 2b shows the generation of measurement sources from an ellipsoid with a lidar sensor using ray tracing. The first step to calculate the intersections is to transform the position of the sensor (x_s, y_s, z_s) into BODY coordinates using:

$$\begin{pmatrix} x'_s\\ y'_s\\ z'_s\end{pmatrix} = R_\varphi^{-1} R_\theta^{-1} R_\psi^{-1} \begin{pmatrix} x_s\\ y_s\\ z_s\end{pmatrix}. \tag{7}$$

Then, a single lidar beam into azimuth direction θ and inclination φ can be described using:

$$x = x'_s + r\cos(\phi)\cos(\theta), \tag{8}$$

$$y = y'_s + r\cos(\phi)\sin(\theta), \tag{9}$$

$$z = z'_s + r\sin(\phi). \tag{10}$$

Inserting (8)-(10) into the ellipsoid's equation (2) leads to

$$\left(\frac{(x'_s + r\cos(\phi)\cos(\theta))^2}{l^2}\right) + \left(\frac{(y'_s + r\cos(\phi)\sin(\theta))^2}{w^2}\right) + \left(\frac{(z'_s + r\sin(\phi))^2}{h^2}\right) - \frac{1}{4} = 0, \tag{11}$$

which can be rearranged to

$$\left(\frac{\cos(\phi)^2\cos(\theta)^2}{l^2} + \frac{\cos(\phi)^2\sin(\theta)^2}{w^2} + \frac{\sin(\phi)^2}{h^2}\right) r^2 + \left(\frac{2x'_s\cos(\phi)\cos(\theta)}{l^2} + \frac{2y'_s\cos(\phi)\sin(\theta)}{w^2} + \frac{2z'_s\sin(\phi)}{h^2}\right) r + \frac{{x'_s}^2}{l^2} + \frac{{y'_s}^2}{w^2} + \frac{{z'_s}^2}{h^2} - \frac{1}{4} = 0 \tag{12}$$

and solved for r since it has a quadratic form. If there is no real solution $r \in \mathbb{R}$, there is no intersection of this lidar beam with the ellipsoid. If there are two solutions, the one with the lower value for r is considered as measurement source. Inserting r into (8)-(10) gives the Cartesian BODY coordinates of the measurement source. Finally, the coordinates are transformed back into ENU coordinates using:

$$\begin{pmatrix} x\\ y\\ z\end{pmatrix} = \begin{pmatrix} x_0\\ y_0\\ z_0\end{pmatrix} + R_\varphi R_\theta R_\psi \begin{pmatrix} x\\ y\\ z\end{pmatrix}. \tag{13}$$

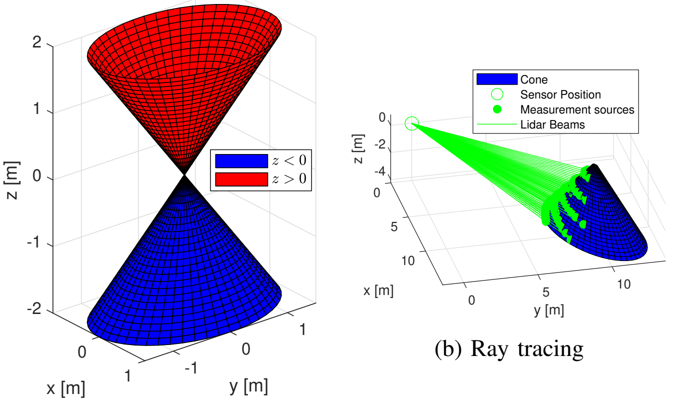
This intersection calculation is performed for each lidar beam.

B. *Elliptical cone*

An elliptical cone's Cartesian representation is given by:

$$\frac{x^2}{\left(\frac{l}{2}\right)^2} + \frac{y^2}{\left(\frac{w}{2}\right)^2} - \frac{z^2}{h^2} = 0. \tag{14}$$

The peak of the cone is placed at the origin, and the lateral surface is not limited. However, for our purposes, we restrict our consideration to the cone's portion with $z \in [-h, 0]$, as illustrated in the blue-highlighted area in Fig. 3a.



(b) Ray tracing

(a) Symmetric representation

Fig. 3: Elliptical cone

The cone can also be expressed parametrically as [16], [30]:

$$c(u, s) = \begin{bmatrix} u \cdot l/2 \cdot \cos(s) \\ u \cdot w/2 \cdot \sin(s) \\ -u \cdot h \end{bmatrix}, \tag{15}$$

with $u \in [0, 1]$ and $s \in [0, 2\pi]$. This parametric form ensures the cone's peak is located at the origin, similar to its Cartesian representation (14). Ray tracing for the elliptical cone follows a methodology similar to that for the ellipsoid. Intersections of lidar beams with the cone's surface are determined by substituting (8)-(10) into the cone equation (14). This yields an equation similar to (12),

$$\left(\frac{\cos(\phi)^2\cos(\theta)^2}{\left(\frac{l}{2}\right)^2} + \frac{\cos(\phi)^2\sin(\theta)^2}{\left(\frac{w}{2}\right)^2} - \frac{\sin(\phi)^2}{h^2}\right) r^2 + \left(\frac{2x'_s\cos(\phi)\cos(\theta)}{\left(\frac{l}{2}\right)^2} + \frac{2y'_s\cos(\phi)\sin(\theta)}{\left(\frac{w}{2}\right)^2} - \frac{2z'_s\sin(\phi)}{h^2}\right) r, + \frac{{x'_s}^2}{\left(\frac{l}{2}\right)^2} + \frac{{y'_s}^2}{\left(\frac{w}{2}\right)^2} - \frac{{z'_s}^2}{h^2} = 0, \tag{16}$$

which is then solved for r. Subsequently, the corresponding Cartesian coordinates are calculated using (8)-(10) and only solutions with $z \in [-h, 0]$ are considered as valid measurement sources, as the cone surface is originally unbounded in z. These valid measurement sources must be transformed into ENU coordinates using (13). Fig. 3b illustrates an example of ray tracing for a 3D elliptical cone. While contour measurements from the elliptical base area could potentially be generated, they have not been incorporated into this work, as they are not visible in the examined scenarios due to observation angle limitations.

C. Rectangular cuboid

The cuboid has a total of six faces, each capable of intersecting with lidar beams to generate measurement sources. These intersection points are also determined in BODY coordinates. Initially, the sensor's position is transformed into this coordinate system using (7). In the BODY coordinate system, the cuboid's six plane equations are defined as follows: $x = \pm\frac{l}{2}$, $y = \pm\frac{w}{2}$ and $= \pm\frac{z}{2}$. To find the intersections with a lidar beam, we substitute (9) into the plane equations and solve for the radius r, leading to:

$$r_{1,2} = \frac{\pm\frac{l}{2} - x'_s}{\cos(\phi)\cos(\theta)}, \tag{17}$$

$$r_{3,4} = \frac{\pm\frac{w}{2} - y'_s}{\cos(\phi)\sin(\theta)}, \tag{18}$$

$$r_{5,6} = \frac{\pm\frac{h}{2} - z'_s}{\sin(\theta)}. \tag{19}$$

The Cartesian coordinates are then obtained using (8)-(10). However, due to the mathematical unboundedness of the planes, only solutions satisfying $-\frac{l}{2} \leq x \leq \frac{l}{2}$, $-\frac{w}{2} \leq y \leq \frac{w}{2}$, and $-\frac{h}{2} \leq z \leq \frac{h}{2}$ are considered as intersections of the lidar beams with the cuboid's contour. Among these solutions, only those with the lower radius r_i are considered as measurement sources, as lidar measurements only occur on the face directly facing the sensor. Finally, measurement sources are transformed into ENU coordinates using (13).

IV. 3D-EXTENDED OBJECT TRACKING WITH VIRTUAL MEASUREMENT MODEL ADAPTATION

The framework for 3D-EOT and shape classification, shown in Fig. 4, builds upon our prior research [23, p. 5], which focused on tracking and classifying 2D objects. The central concept involves adapting a VMM (see Fig. 1) for each shape to obtain the results $\tilde{\underline{u}}_{k,<\text{shape}>}$ and $\tilde{\boldsymbol{Z}}_{k,<\text{shape}>}$. The place-holder $<$ shape $>$ is then replaced with ell, con or cub. For example, the vector $\tilde{\underline{u}}_{k,ell}$ contains the center of gravity and the extension parameters that would cause a set of virtual measurement $\tilde{\boldsymbol{Z}}_{k,ell}$ with the ellipsoidal VMM, that has identical mean and covariance as the real measurements, filtered with random matrices. The same applies to $\tilde{\underline{u}}_{k,con}$ and $\tilde{\underline{u}}_{k,cub}$ for the cone and cuboid, respectively.

In the subsequent stage of shape classification, real and virtual measurements are compared for each shape hypothesis to determine the best-fitting shape. This paper introduces four significant enhancements compared to our previous work:

- Expansion to 3D (see Sec. III)
- Eigenvector matching
- Improved cost function
- Shape classification using Chamfer distances

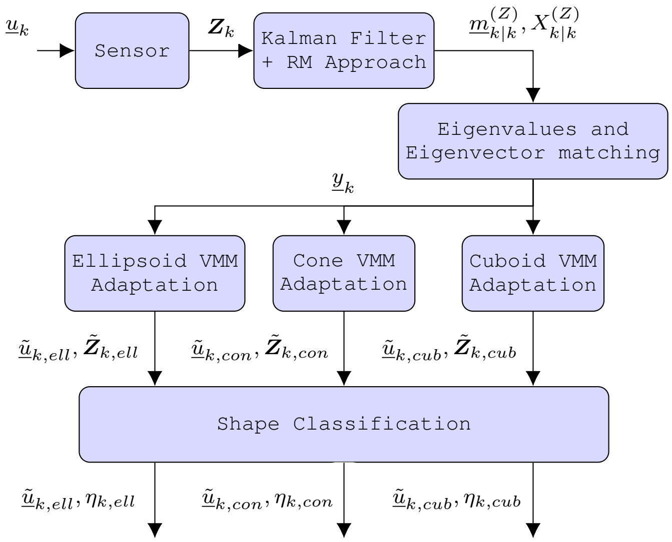


Fig. 4: Extended object tracking and shape classification with three virtual measurement models. For each VMM, an adaptation loop as shown as in Fig. 1 is executed. Fig. adapted from [23, p. 5].

A. Eigenvector matching

In previous work on VMMs [5]–[7], [23], it was assumed that the larger eigenvalue corresponds to the length of the object and the smaller eigenvalue corresponds to the width. However, in scenarios where the difference between these values is small, there's a risk that the tracking algorithm may misinterpret l and w. This challenge becomes more pronounced in 3D extended object tracking (EOT) with the introduction of height as a third parameter. An alternative approach that remains independent of extension parameters is eigenvector-based matching, outlined in Algorithm 1.

Algorithm 1 Eigenvector-based matching

```
1. Calculate eigenvalues and eigenvectors of $X^{(Z)}_{k|k}$.
2. Identify the eigenvector with the highest z-component and assign the corresponding eigenvalue to height.
3. Calculate yaw angles from the remaining eigenvalues.
4. Calculate yaw angles from kinematics.
5. Assign the eigenvalue where the yaw angle of the eigenvector is closest to the kinematic yaw angle to length.
6. Assign the remaining eigenvalue to width.
```

It's important to note that this approach works effectively for small roll and pitch angles. However, for large roll and pitch angles, the eigenvalue with the highest z-component in its eigenvector may not necessarily correspond to height.

B. Improved cost function

In scenarios where the sensor's observation angle does not ensure full observability of all extension parameters, there's a risk of undesired increases in the extension estimation, even if the error ϵ is minimized during adaptation. To address this issue and maintain low extension parameters, we propose replacing the error term ϵ with a modified cost function: $c = \epsilon + \left(\tilde{l} + \tilde{w} + \tilde{h}\right)/\eta_c$, where η_c represents a normalizing constant. By minimizing this cost function during adaptation, we aim to obtain the smallest extension parameters that remain aligned with the measurements.

C. Shape classification using Chamfer distances

Previously, shape classification within the VMM framework relied on comparing histograms of real measurements $\boldsymbol{Z}_k$ with those of artificial measurements for each shape hypothesis [23]. However, this method did not consistently achieve high classification quality, especially in scenarios with moderate to high measurement noise [23].

To enhance classification accuracy, we introduce the use of the Chamfer distance as a metric to evaluate the similarity between the two sets of measurements. The Chamfer distance between sets A and B, denoted as $d_C(A, B)$, is computed using (1). The classification scores for each shape hypothesis are then determined as follows:

$$\eta_{k,ell} = d_C\left(\tilde{\boldsymbol{Z}}_{k,ell}, \boldsymbol{Z}_k\right), \tag{20}$$

$$\eta_{k,con} = d_C\left(\tilde{\boldsymbol{Z}}_{k,con}, \boldsymbol{Z}_k\right), \tag{21}$$

$$\eta_{k,cub} = d_C\left(\tilde{\boldsymbol{Z}}_{k,cub}, \boldsymbol{Z}_k\right). \tag{22}$$

The shape with the lowest score is considered as the estimated shape. To improve accuracy further, real measurements from multiple time steps are accumulated using a sliding window approach with a window length L. These accumulated measurements are then transformed into the BODY coordinate system of the VMM's state estimation and compared with the virtual measurements generated by the corresponding VMM.

V. RESULTS

To evaluate the 3D-EOT approach, we use both simulation studies and real lidar data.

A. Simulation studies with an ellipsoidal target

1) Tracking: For the simulation studies, a constant acceleration (CA) model with $\underline{x} = (x, y, z, v_x, v_y, v_z, a_x, a_y, a_z)^\top$, affected by white Gaussian process noise with $\sigma_{wx} = \sigma_{wy} = \sigma_{wz} = 0.5$ m/s^4, was used to generate a reference trajectory. The sampling period was $T = 0.1$ s and the total number of timesteps was $k_{max} = 280$. Fig. 5 shows the trajectory. Measurements were generated by ray tracing as proposed in Sec. III. The sensor, located at the origin, had vertical and horizontal resolutions of $1°$. Measurements were affected by white Gaussian noise with $\sigma_x = \sigma_y = \sigma_z = 0.5$ m.

The tracking results for the ellipsoid are shown in Fig. 5. The linear Kalman filter with random matrix extension estimation is represented in black. The 1-σ-ellipse of the filtered covariance matrix is barely visible as it lies within the reference ellipsoid. The estimated CoG-Trajectory shows a clear bias toward the sensor compared to the object's center. However, the RM approach was not made for measurement models using ray tracing but is required for the following VMM adaptation. The VMM adaptation is achieved with 2 iterations per time step. The assignment of eigenvalues to extension parameters is performed using the eigenvector assignment as proposed in Alg. 1, however, the improved cost function (see Sec. IV-B) was not necessary for this scenario and therefore not used. The results after VMM adaptation, shown in Fig. 5, precisely match the reference trajectory without any significant bias.

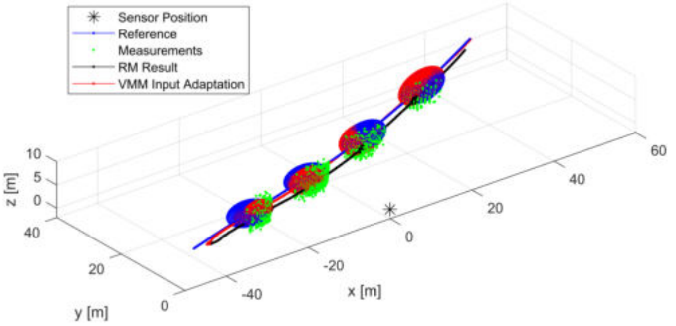


Fig. 5: Tracking results for the ellipsoid. Extension estimation and measurement shown every 60 time steps.

To ensure robustness and reliability, we performed a Monte Carlo simulation with 100 runs, wherein process noise and measurement noise were sampled independently in each run. The mean and standard deviation of the extension estimation from all runs are presented in Fig. 6 over time and in Table I.

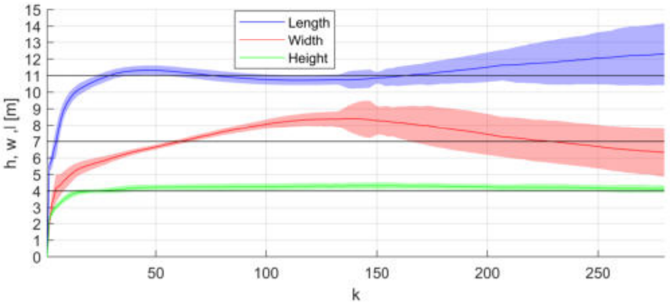


Fig. 6: MC results for the ellipsoid.

TABLE I: Mean and std. deviation from 100 MC runs.

	Reference	Ellipsoid	Cone	Cuboid
length [m]	11	11.1 ± 1.4	12.9 ± 2.6	7.3 ± 0.9
width [m]	7	7.1 ± 1.4	6.2 ± 2.3	4.2 ± 1.8
height [m]	4	4.2 ± 0.4	4.7 ± 0.9	3.3 ± 0.3

The height estimation yielded the best results, benefitting from an optimal observation angle from the sensor to the object. Toward the end of the scenario, the length estimation faced challenges as the object could only be observed from behind. When the object is close to the sensor but only seen from the side, the width estimation becomes challenging.

2) Classification: In scenarios where the ellipsoidal shape is initially unknown and must be determined, the cone VMM and the cuboid VMM are also employed simultaneously. Their results are presented in Table I and are notably inferior as they assume the wrong shape. However, the artificial measurements of each VMM $\tilde{\boldsymbol{Z}}_k$ can be compared to the real measurements $\boldsymbol{Z}_k$ using Chamfer distances to identify the correct shape. Figure 7 shows the Chamfer distances for a window-length of $L = 5$, averaged over 100 Monte Carlo runs. On average, the Chamfer distance of the virtual measurements of the ellipsoid to the real measurements is the smallest, indicating correct classification. The $1 - \sigma$ ranges demonstrate highly reliable correct classification overall, but overlap toward the end, suggesting occasional misclassifications in some Monte Carlo runs. However, these misclassifications were rare: the ellipsoid was correctly classified in 97.9% of all time steps, while the cuboid had the lowest Chamfer distance in 1.9% of cases, and the cone in only 0.2%.

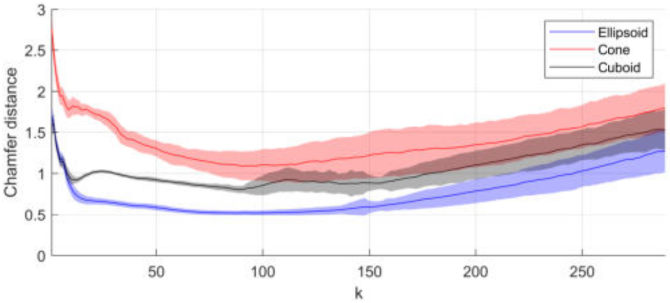
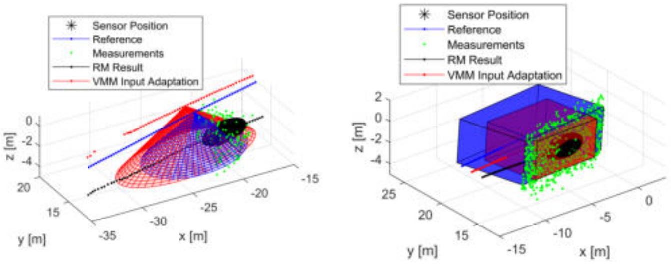


Fig. 7: Chamfer distances for three VMMs. MC results.

B. Simulation studies with a cone and cuboid

The same scenario is simulated, but with different reference shapes. Fig. 8 shows the results for the cone and cuboid at time step $k = 60$. Overall, similar tracking results were obtained for the cone as for the ellipsoid. However, for the cuboid, tracking results were significantly degraded in some time steps. There was notable underestimation of the extension, particularly when only one surface was visible, leading to observability issues (see, e.g., Figure 8b). The classification results for all shapes are summarized in Table II, determined through 100 Monte Carlo (MC) runs per shape. While the ellipsoid and cone could be reliably classified correctly, the cuboid was often mistakenly classified as an ellipsoid.



(a) Cone

(b) Cuboid

Fig. 8: Tracking results at $k = 60$.

TABLE II: Classification results from 100 MC runs per shape.

	classified as		
Reference	Ellipsoid	Cone	Cuboid
Ellipsoid	97.9%	0.2%	1.9%
Cone	2.9%	94.7%	2.4%
Cuboid	46.6%	0.1%	53.3%

C. Real-world scenario with a sailing boat

To evaluate the EOT approach in a real-world setting, we used a recording from the Lake of Constance (see Fig. 9), captured by a Velodyne Alpha Prime (VLP-128) lidar. The lidar was mounted on a catamaran traveling at approximately 32 km/h from Friedrichshafen to Constance. To compensate the ego motion, we used a differential GPS for position and yaw, along with an IMU for roll and pitch measurements. The camera image in Fig. 9 provides an overview of the real-world scenario. Fig. 10 shows the lidar measurements, the random matrix result and the extension estimation after the VMM adaptation for a single sailing boat. Our algorithm selected an elliptical cone as the best-fitting shape.

Fig. 9: Camera image of the real-world scenario.

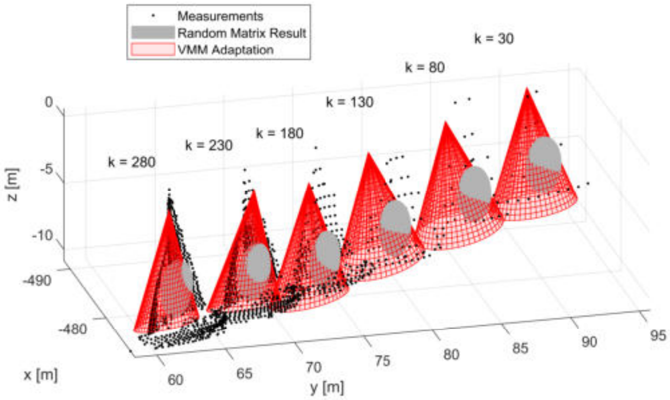
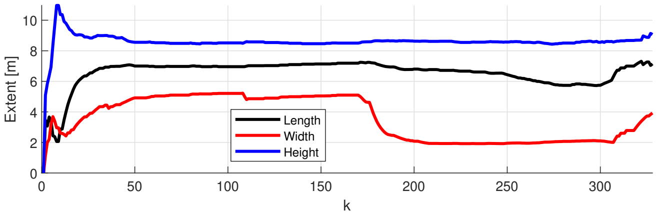


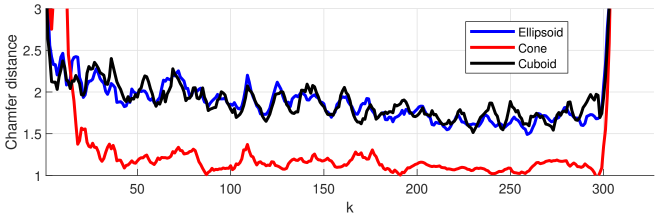
Fig. 10: Extension estimation of a sailing boat using a VMM with elliptical cone.

At the beginning of the scenario, the distance of the sailing boat is ≈ 240 m, therefore the number of measurements is low. As the catamaran approaches the sailing boat, the number of measurements steadily increases. By the end of the scenario, the distance is approximately 70 meters.

Fig. 11a shows the estimated extension parameters over k. The length and height estimates appear plausible and stay almost constant throughout the scenario. However, the width estimation presents challenges attributed to several factors:



(a) Extension estimation (Cone-VMM).



(b) Chamfer distances

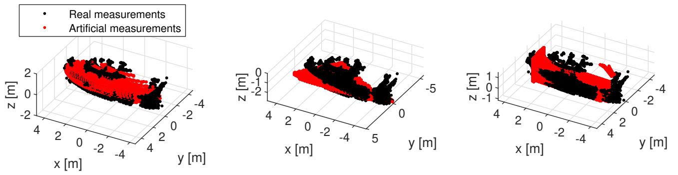
Fig. 11: Tracking results from the sailing boat.

Firstly, due to the direction of the wind, the sail is positioned on the backside of the boat. With ray tracing, measurements are assumed to originate from the front, resulting in a modeling mismatch. Secondly, although the boat tilts slightly, neglecting roll and pitch results in an always upright estimated cone. Thirdly, since the object is consistently observed from the side, estimating width is inherently more complex than estimating length and height.

The classification results given, by chamfer distances, are shown in Fig. 11b. Apart from a few time steps at the beginning, when the Kalman filter has not yet converged, and at the end, the cone consistently achieves the lowest chamfer distance. This outcome is intuitive, as a cone visually seems to be the best fit for a sailing boat (see Fig. 11).

D. Real-world scenario with a motor boat

In the second real-world scenario, lidar data from a motorboat, the "Solgenia," are used. For the "Solgenia", both a CAD model and reference extension parameters are available. Fig. 13 shows the measurements, the tracking results after VMM adaptation, and the CAD model (manually fitted to measurements). Length and width of the "Solgenia" can be estimated precisely with an error of only 10 cm. This precision results from the fact that the base, resp. 2D-top view, of the "Solgenia" is almost a perfect ellipse. However, the height is somewhat overestimated, particularly towards the bottom, where the estimated ellipsoid is significantly further from the CAD model. Regarding the height parameter, the elliptical assumption is not entirely met, posing a challenge to accurate estimation. Since the shape is not known in advance, our approach aims to classify the shape as well. In our classification process, we accumulate measurements in a window of $L = 5$. As an example, Fig. 12 shows the real and artificial measurements for the time steps $k = 10$ up to $k = 15$. It is already clearly visible that the point clouds Fig. 12a are the most similar. Table III shows the mean Chamfer distances and the empirical probabilities that the corresponding shape has the lowest Chamfer distance. In 91.4% of all time steps, the ellipsoid was the best fitting shape since it had the lowest Chamfer distance.



(a) Ellipsoid (b) Cone (c) Cuboid

Fig. 12: Real vs. artificial measurements for classification.

TABLE III: Classification results for the motor boat

	Ellipsoid	Cone	Cuboid
Mean d_C	0.50	0.67	0.59
classified as ... [%]	91.4	4.8	3.8

VI. CONCLUSION

In this paper, we have extended the VMM concept to 3D tracking and derived the necessary measurement models. These are ray tracing methods that provide analytical solutions for the intersection points of the lidar beams with the surface of the object. Our study highlights the effectiveness of improved classification techniques, particularly the use of Chamfer distances, which yielded remarkable results in both simulation and real-world lidar data scenarios. Notably, our algorithm demonstrated the classification of a sailing boat as a cone and a motorboat as an ellipsoid. Knowing the right shape model leads to precise extension estimation, particularly in length and height, for the ellipsoidal motorboat also in width. Challenges in width estimation for the sailing boat and in some simulation studies, primarily attributed to observation angles, were identified and underscored the ongoing need for future work and improvements. Looking ahead, optimizing computational efficiency remains a priority for future research. Particularly, addressing inefficiencies in cone and cuboid ray tracing methods, where the initially unrestricted surfaces result in numerous rejected intersections, will be essential for enhancing real-time tracking performance.

REFERENCES

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

[2] 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.

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

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

[5] P. Hoher, S. Wirtensohn, T. Baur, J. Reuter, F. Govaers, and W. Koch, "Extended target tracking with a lidar sensor using random matrices and a virtual measurement model," *IEEE Transactions on Signal Processing*, vol. 70, pp. 228–239, 2022.

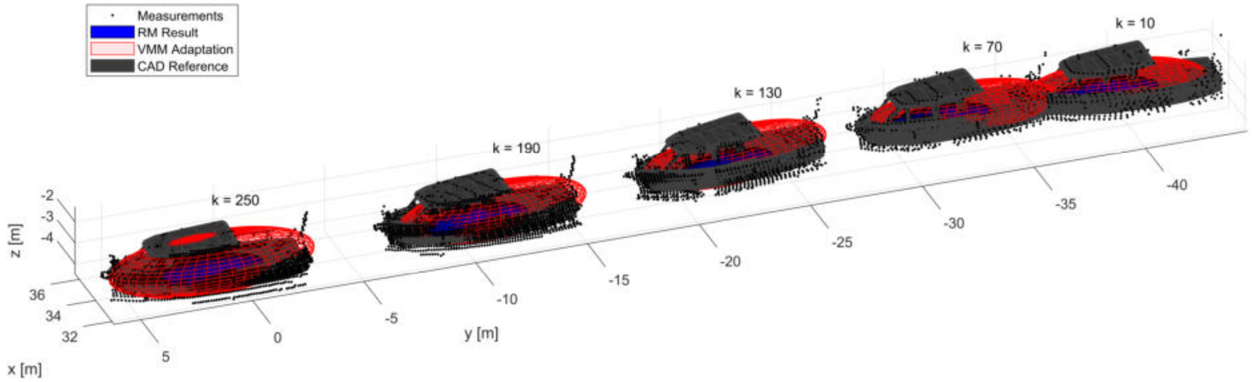


Fig. 13: Tracking results for the motorboat "Solgenia". Reference parameters: $l = 8.4$ m, $w = 2.5$ m, $h = 2.0$ m. Estimated parameters (mean): $\tilde{l} = 8.5$ m, $\tilde{w} = 2.4$ m, $\tilde{h} = 2.5$ m.

[6] P. Hoher, J. Reuter, F. Govaers, and W. Koch, "Tracking of partially visible elliptical objects with a lidar sensor using random matrices and a virtual measurement model," in *2022 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, 2022, pp. 1–6.

[7] P. Hoher, J. Reuter, D. Dold, D. Griesser, F. Govaers, and W. Koch, "Extended target tracking with a lidar sensor using random matrices and a Gaussian processes regression model," in *2023 IEEE 26th International Conference on Information Fusion (FUSION)*, 2023, pp. 1–8.

[8] H. Alqaderi, F. Govaers, and R. Schulz, "Spacial elliptical model for extended target tracking using laser measurements," in *2019 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, 2019, pp. 1–6.

[9] K. Granström, C. Lundquist, and U. Orguner, "Tracking rectangular and elliptical extended targets using laser measurements," in *14th International Conference on Information Fusion*, 2011, pp. 1–8.

[10] K. A. Ruud, E. F. Brekke, and J. Eidsvik, "Lidar extended object tracking of a maritime vessel using an ellipsoidal contour model," in *2018 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, 2018, pp. 1–6.

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

[12] K. Granström, S. Reuter, D. Meissner, and A. Scheel, "A multiple model PHD approach to tracking of cars under an assumed rectangular shape," in *17th International Conference on Information Fusion (FUSION)*, 2014, pp. 1–8.

[13] M. Baum and U. D. Hanebeck, "Tracking an extended object modeled as an axis-aligned rectangle," in *4th German Work- shop on Sensor Data Fusion: Trends, Solutions, Applications (SDF 2009)*, 2009.

[14] T. Hirscher, A. Scheel, S. Reuter, and K. Dietmayer, "Multiple extended object tracking using Gaussian processes," in *2016 19th International Conference on Information Fusion (FUSION)*, 2016, pp. 868–875.

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

[16] T. Baur, J. Reuter, A. Zea, and U. D. Hanebeck, "Extent estimation of sailing boats applying elliptic cones to 3D lidar data," in *2022 25th International Conference on Information Fusion (FUSION)*, 2022, pp. 1–8.

[17] F. Faion, M. Baum, and U. D. Hanebeck, "Tracking 3D shapes in noisy point clouds with random hypersurface models," in *2012 15th International Conference on Information Fusion*, 2012, pp. 2230–2235.

[18] P. Hoher, J. Reuter, F. Govaers, and W. Koch, "Joint parameter estimation and trajectory tracking of bounding boxes," in *2021 IEEE 24th International Conference on Information Fusion (FUSION)*, 2021, pp. 1–8.

[19] X. Li and J. E. Guivant, "Efficient and accurate object detection with simultaneous classification and tracking under limited computing power," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 6, pp. 5740–5751, 2023.

[20] M. Malmström, A. Kullberg, I. Skog, D. Axehill, and F. Gustafsson, "Extended target tracking utilizing machine-learning software–with applications to animal classification," *IEEE Signal Processing Letters*, vol. 31, pp. 376–380, 2024.

[21] C. Magnant, S. Kemkemian, and L. Zimmer, "Joint tracking and classification for extended targets in maritime surveillance," in *2018 IEEE Radar Conference (RadarConf18)*, 2018, pp. 1117–1122.

[22] L. Wang and R. Zhan, "Joint detection, tracking, and classification of multiple maneuvering star-convex extended targets," *IEEE Sensors Journal*, vol. 24, no. 4, pp. 5004–5024, 2024.

[23] P. Hoher, J. Reuter, F. Govaers, and W. Koch, "Extended object tracking and shape classification using random matrices and virtual measurement models," in *2023 IEEE Symposium Sensor Data Fusion and International Conference on Multisensor Fusion and Integration (SDF-MFI)*, 2023, pp. 1–8.

[24] X. Zhang and J. Lan, "Measurement combination estimator for multisensor extended object tracking using random matrix," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 1, pp. 698–715, 2024.

[25] Q. Jiao and X. Yang, "Distributed variational measurement update for extended target tracking with random matrix," *IEEE Transactions on Aerospace and Electronic Systems*, pp. 1–15, 2024.

[26] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf, "Parametric correspondence and Chamfer matching: Two new techniques for image matching," in *Proceedings of the 5th International Joint Conference on Artificial Intelligence - Volume 2*, ser. IJCAI'77, Cambridge, USA: Morgan Kaufmann Publishers Inc., 1977, 659–663.

[27] T. Wu, L. Pan, J. Zhang, T. Wang, Z. Liu, and D. Lin, "Balanced Chamfer distance as a comprehensive metric for point cloud completion," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34, Curran Associates, Inc., 2021, pp. 29 088–29 100.

[28] F. Lin, Y. Yue, Z. Zhang, *et al.*, "Infocd: A contrastive Chamfer distance loss for point cloud completion," in *Advances in Neural Information Processing Systems*, A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36, Curran Associates, Inc., 2023, pp. 76 960–76 973.

[29] T. Fossen, *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*. Marine Cybernetics, 2002.

[30] F. Faion, A. Zea, J. Steinbring, M. Baum, and U. D. Hanebeck, "Recursive Bayesian pose and shape estimation of 3D objects using transformed plane curves," in *2015 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, 2015, pp. 1–6.