Tracking Arbitrarily Shaped Extended Objects Using Gaussian Processes

Murat Kumru
Vehicle Motion and Thermal Management
Volvo Group Trucks Technology
Gothenburg, Sweden
murat.kumru@volvo.com

Emre Özkan
Department of Electrical and Electronics Engineering
Middle East Technical University
Ankara, Turkey
emreo@metu.edu.tr

***Abstract*—In this paper, we consider the problem of tracking dynamic objects with unknown shapes using point cloud measurements generated by sensors such as lidars and radars. Specifically, our objective is to extend the Gaussian process-based extended object tracking (GPEOT) framework to encompass a broader class of objects. The derivation of the existing GPEOT algorithms is based on the assumption that the object of interest is star-convex. This assumption enables the modeling of the object's extent through a radial distance function, which is described by a Gaussian process (GP). To enhance the flexibility of the resulting trackers, we propose the utilization of a potential function to indicate the unknown object extent. This approach enables the representation of objects with arbitrary shapes, including those that are non-convex and composed of disconnected components. Closely following the original formulation of GPEOT, the potential function is then modeled by a GP, which systematically accounts for the intrinsic spatial correlation of the extent. Furthermore, we develop a state-space model that incorporates both kinematic variables and an approximate description of the underlying GP model. The state vector can be estimated via a standard Bayesian technique, leading to an EOT algorithm. Through simulation experiments, we demonstrate the suggested method can satisfactorily estimate the kinematic variables of the objects while simultaneously learning their complex shapes.**

***Index Terms*—Extended Object Tracking, Gaussian Processes**

I. Introduction

Conventional object tracking is based on the point-source model that assumes an object can generate at most one measurement per scan, [1]. Recent improvements in sensor technology have resulted in various applications in which a single object can produce multiple measurements in each scan. A well-established research area, known as *extended object tracking* (EOT), systematically addresses this technical problem. The resulting methods aim to estimate the object's extent along with the kinematic variables. Numerous successful solutions have been developed, using a variety of formulations, e.g., [2]–[13]. An extensive review of the current is available in [14] and [15].

This paper focuses on the Gaussian process-based extended object tracking (GPEOT) approach, where the unknown extent is modeled by a Gaussian process (GP), and it is estimated simultaneously with the kinematics using sequentially available measurements.

Following its introduction in [16], GPEOT and its variants have been adopted in several studies; see, for example, [17]–[26]. A critical shortcoming of the current GPEOT literature is that the object of interest is assumed to be star-convex which impedes the widespread use of the method in various tracking applications. Drawing upon this observation, in this paper, we propose an algorithm that simultaneously estimates the kinematics and learns an arbitrarily shaped object's extent without imposing any convexity assumption.

Seeking a general extent description, we hereby suggest modeling the unknown object extent by a scalar-valued potential function. The potential value of a point in space is used to indicate whether the point is included by the object's extent or not. With this formulation, it is possible to represent arbitrarily shaped objects, which might be non-convex and consist of disconnected subparts. The rest of the derivation closely follows the existing GPEOT models. In particular, we model the latent potential function by a GP, which is approximated at some inducing points. Subsequently, a unified state-space model, which consists of both the kinematic variables and the approximate description of the GP model, is constructed. Regarding this state-space model, we implement an extended Kalman filter (EKF) that efficiently estimates the state vector by recursively processing point cloud measurements.

II. Proposed Method

A. Extent Representation

In this section, we will reveal the representation of the object extent used in the derivation. While formulating the description, our objective is to attain high representational power, which will allow the resulting method to generalize to a broad class of object shapes, and be sufficiently compact for an online tracking application.

In this regard, we model the object extent by a scalar potential function, i.e., $f(\mathbf{u}) : \mathbb{R}^d \to \mathbb{R}$, where $\mathbf{u} \in \mathbb{R}^d$ indicates a point in the coordinate frame. More specifically, we follow the conventional denotation and define the potential function as

$$f(\mathbf{u}) = \begin{cases} +1, & \mathbf{u} \in \mathcal{S} \\ -1, & \mathbf{u} \notin \mathcal{S} \end{cases}, \tag{1}$$

where $\mathcal{S}$ denotes the set of all points contained within the object extent. $f(\mathbf{u})$ can be interpreted as a descriptor function whose level sets imply the latent object extent. A similar approach has been widely utilized to characterize the occupancy information in the applications of environmental exploration and mapping, see, for example, [27]–[29].

This extent model is of critical importance for the derivation of a tracking algorithm that can learn the extent of the arbitrarily shaped objects. The model does not impose any assumption upon the shape of the object, such as star-convexity. Note that the proposed model is able to express objects consisting of multiple disconnected subparts.

In the upcoming sections, we will first introduce a probabilistic model for the unknown potential function, which later will be translated into a unified state-space model for joint estimation of the kinematics and the extent of the object.

B. Gaussian Process Modeling of Object Extent

Being a non-parametric model, GP establishes a convenient basis for probabilistic modeling of arbitrary potential functions and hence the corresponding object extents. In this regard, the potential function $f(\mathbf{u})$ is assumed to have a GP prior, i.e., $f(\mathbf{u}) \sim \mathcal{GP}(0, k(\mathbf{u}, \mathbf{u}'))$, where $k(\mathbf{u}, \mathbf{u}')$ is the covariance function, which indicates the covariance between the function values evaluated at $\mathbf{u}$ and $\mathbf{u}'$, i.e., $k(\mathbf{u}, \mathbf{u}') = \mathbb{E}[f(\mathbf{u})f(\mathbf{u}')]$.

In a typical object tracking setting, the problem is to estimate the latent variables by processing measurements that are available in a sequential fashion. As the number of measurements monotonically increases over time, a naive implementation of a GP model is infeasible due to the ever-growing storage and computational requirements. Consequently, we hereby employ a recursive approximation of the GP model, which brings in favorable computational properties.

C. Recursive Gaussian Process Regression

We rely on an approximation of a GP that summarizes the original model at a finite set of basis points, i.e., $\mathbf{f} \triangleq [f(u_1^{\mathbf{f}}) \ \dots \ f(u_{N^{\mathbf{f}}}^{\mathbf{f}})]^\top$ evaluated at the inputs $\mathbf{u}^{\mathbf{f}} \triangleq [u_1^{\mathbf{f}} \ \dots \ u_{N^{\mathbf{f}}}^{\mathbf{f}}]^\top$. More specifically, assuming $\mathbf{f}$ provides the sufficient statistics for the observations, we will be able to compute the posterior distribution $p(\mathbf{f}|y_{1:N})$ recursively. To this end, we refer to the underlying GP model to offer the measurement likelihood function and the prior density. The joint distribution of an arbitrary function value $f_k = f(u_k)$ and the basis points $\mathbf{f}$ is revealed by the GP as

$$\begin{bmatrix} f_k \\ \mathbf{f} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} k(u_k, u_k) & K(u_k, \mathbf{u}^{\mathbf{f}}) \\ K(\mathbf{u}^{\mathbf{f}}, u_k) & K(\mathbf{u}^{\mathbf{f}}, \mathbf{u}^{\mathbf{f}}) \end{bmatrix}\right). \tag{2}$$

Thereafter, following the standard GP regression equations, we can write

$$p(f_k|\mathbf{f}) = \mathcal{N}(f_k; H_k^{\mathbf{f}}\mathbf{f}, R_k^{\mathbf{f}}), \tag{3a}$$
$$p(\mathbf{f}) = \mathcal{N}(\mathbf{0}, P_0^{\mathbf{f}}), \tag{3b}$$

where

$$H_k^{\mathbf{f}} = H^{\mathbf{f}}(u_k) = K(u_k, \mathbf{u}^{\mathbf{f}})[K(\mathbf{u}^{\mathbf{f}}, \mathbf{u}^{\mathbf{f}})]^{-1}, \tag{3c}$$
$$R_k^{\mathbf{f}} = R^{\mathbf{f}}(u_k) = k(u_k, u_k) - K(u_k, \mathbf{u}^{\mathbf{f}}) [K(\mathbf{u}^{\mathbf{f}}, \mathbf{u}^{\mathbf{f}})]^{-1} K(\mathbf{u}^{\mathbf{f}}, u_k), \tag{3d}$$
$$P_0^{\mathbf{f}} = K(\mathbf{u}^{\mathbf{f}}, \mathbf{u}^{\mathbf{f}}). \tag{3e}$$

Regarding (3), we can build the equivalent state-space model, which will be used within a Kalman filter for recursive inference, [16].

$$\mathbf{f}_{k+1} = \mathbf{f}_k, \tag{4a}$$
$$f_k = H_k^{\mathbf{f}}\ \mathbf{f}_k + e_k^{\mathbf{f}}, \quad e_k^{\mathbf{f}} \sim \mathcal{N}(0, R_k^{\mathbf{f}}), \tag{4b}$$
$$\mathbf{f}_0 \sim \mathcal{N}(\mathbf{0}, P_0^{\mathbf{f}}), \tag{4c}$$

where $\mathbf{f}_k$ is defined as the latent function values evaluated at the predetermined inputs, i.e., $\mathbf{f}_k \triangleq \mathbf{f}$.

D. State-Space Model

We construct a state-space model considering both the kinematics and the extent of the object. The state vector is defined as $\mathbf{x}_k \triangleq \left[\bar{\mathbf{x}}_k^\top \ \mathbf{f}_k^\top\right]^\top$, where $\bar{\mathbf{x}}_k$ denotes the kinematic variables consisting of the object position $\mathbf{c}_k$, orientation ψ_k, and the corresponding time derivatives, $\mathbf{x}_k^*$, i.e., $\bar{\mathbf{x}}_k \triangleq \left[\mathbf{c}_k^\top \ \psi_k \ \mathbf{x}_k^{*\top}\right]^\top$; $\mathbf{f}_k$ indicates the parameterized description of the potential function.

1) Measurement Model: In this section, we derive a measurement model that reveals the relation between the state variables and the observations.

$\mathcal{Y}_k = \{y_{k,l}\}_{l=1}^{n_k}$ denotes the set of point measurements collected at time k. In this study, we assume that a point measurement $y_{k,l}$ encapsulates two types of information: (i) the detection point in two-dimensional space $\mathbf{p}_{k,l} \in \mathbb{R}^2$, (ii) the associated occupancy/free-space information $o_{k,l}$. We designate $o_{k,l}$ as a binary-valued variable, i.e., $o_{k,l} \in \{-1, +1\}$, in accordance with the definition of the potential function in (1). In particular, $o_{k,l}$ takes the value of $+1$ for occupancy measurements, or it is -1 for free-space measurements. Consequently, a single measurement is specified by the tuple $y_{k,l} = (\mathbf{p}_{k,l}, o_{k,l})$.

In practice, such measurements can be generated by various sensor modalities, such as lidars, radars, and depth cameras. In all these cases, the measurements provide some noisy observations of the true source points that give rise to the sensor readings. Accordingly, we relate $o_{k,l}$ to the underlying potential function evaluated at $\mathbf{p}_{k,l}^L$ as in

$$o_{k,l} = f(\mathbf{p}_{k,l}^L(\mathbf{c}_k, \psi_k)) + \bar{e}_{k,l}, \quad \bar{e}_{k,l} \sim \mathcal{N}(0, R) \tag{5}$$

$\bar{e}_{k,l}$ is assumed to be zero-mean Gaussian measurement noise with covariance R, and the expression for $\mathbf{p}_{k,l}^L$ is given as

$$\mathbf{p}_{k,l}^L(\mathbf{c}_k, \psi_k) = R(\psi_k)(\mathbf{p}_{k,l} - \mathbf{c}_k). \tag{6}$$

where $\mathbf{p}_{k,l}$ is the corresponding point resolved in the global coordinate frame, and $R(\psi_k)$ is the rotation matrix between the global and the local frames defined as

$$R(\psi_k) = \begin{bmatrix} \cos(\psi_k) & \sin(\psi_k) \\ -\sin(\psi_k) & \cos(\psi_k) \end{bmatrix}. \tag{7}$$

We illustrate the variables used in the measurement model in Fig. 1.

By plugging in the GP representation of the potential function, we can rewrite the measurement equation as

$$\begin{aligned} 0 &= \underbrace{-o_{k,l} + H^{\mathbf{f}}\left(\mathbf{p}_{k,l}^{L}(\mathbf{c}_k, \psi_k)\right)\, \mathbf{f}_k}_{=h_{k,l}(\mathbf{x}_k, y_{k,l})} + \underbrace{e_{k,l}^{\mathbf{f}} + \bar{e}_{k,l}}_{=e_{k,l}} \\ &= h_{k,l}(\mathbf{x}_k, y_{k,l}) + e_{k,l}, \quad e_{k,l} \sim \mathcal{N}(0, R_{k,l}). \end{aligned} \tag{8a}$$

where

$$R_{k,l} = R_{k,l}^{\mathbf{f}} + R. \tag{8b}$$

The model (8) is nonlinear and implicit, where the measurement cannot be written explicitly as a function of the state variables and the noise term. This type of model has been commonly used in the literature to derive recursive Bayesian filters, e.g., [20], [21]. Note that this approach, being an example of greedy association models, can potentially lead to biased estimates in high-noise scenarios, [14].

2) Process Model: To complete the definition of the state-space model, we formulate the time evolution of the state vector by the following process model.

$$\mathbf{x}_{k+1} = F\mathbf{x}_k + \mathbf{w}_k, \quad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, Q_k), \tag{9}$$

where F is the system matrix, and $\mathbf{w}_k$ is the process noise, which is assumed to be Gaussian with zero mean and covariance Q_k. Suppose that the dynamics of the kinematic and the extent variables do not interact with each other, then we can partition the process model as

$$F = \begin{bmatrix} \bar{F} & 0 \\ 0 & F^{\mathbf{f}} \end{bmatrix}, \; Q_k = \begin{bmatrix} \bar{Q} & 0 \\ 0 & Q_k^{\mathbf{f}} \end{bmatrix}. \tag{10}$$

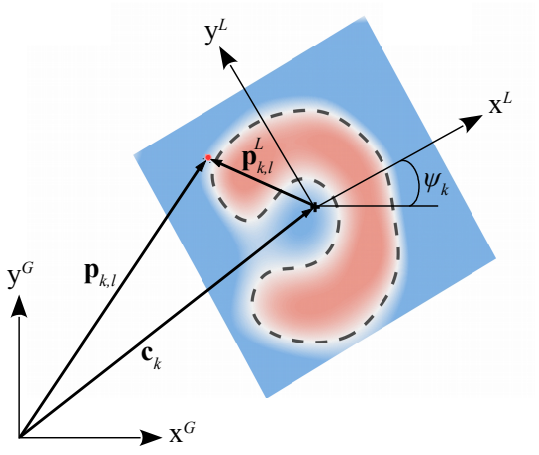


Fig. 1: Illustration of the variables included in the measurement model. The red circle indicates the point measurement. The dashed curve denotes the object of interest.

The formulation does not put any restrictions on the selection of the process model. We hereby employ the nearly constant velocity model for the object kinematics, i.e.,

$$\bar{F} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \otimes I_3, \; \bar{Q} = \begin{bmatrix} \frac{T}{3}^3 & \frac{T}{2}^2 \\ \frac{T}{2}^2 & T \end{bmatrix} \otimes \begin{bmatrix} \sigma_c^2 & 0 & 0 \\ 0 & \sigma_c^2 & 0 \\ 0 & 0 & \sigma_\psi^2 \end{bmatrix}, \tag{11}$$

where σ_c^2 and σ_ψ^2 are the process noise variances for the position and the orientation, respectively; T is the sampling time; and $\otimes$ denotes the Kronecker product.

For the object extent, we utilize the following dynamic model, which is known to attain the predicted distribution with maximum entropy for unknown and slowly varying processes [30, Therorem 1],

$$F^{\mathbf{f}} = I, \; Q_k^{\mathbf{f}} = \left(\lambda^{-1} - 1\right) P_{k|k}^{\mathbf{f}}, \tag{12}$$

where I indicates the identity matrix, $P_{k|k}^{\mathbf{f}}$ is the covariance of the estimated extent. Notice that this model basically scales up the prediction covariance as $P_{k+1|k}^{\mathbf{f}} = \lambda^{-1} P_{k|k}^{\mathbf{f}}$ for $\lambda < 1$, while the mean of the prediction is the same with that of the estimated density.

3) Inference: In this section, we present an effective inference method by relying on the state-space model developed earlier. Particularly, the objective is to propose a method that recursively computes the posterior distribution of the state vector. To this end, we will utilize an extended Kalman filter (EKF) considering the nonlinearities in the measurement model.

To incorporate multiple measurements $\{y_{k,l}\}_{l=1}^{n_k}$ available at time k, we first form a measurement vector by concatenating the individual measurements,

$$\mathbf{y}_k = \left[y_{k,1}, \; \ldots, \; y_{k,n_k}\right]^\top. \tag{13}$$

The corresponding measurement model can be written as

$$\mathbf{0} = \mathbf{h}(\mathbf{x}_k, \mathbf{y}_k) + \mathbf{e}_k, \quad \mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, R_k), \tag{14}$$

where

$$\mathbf{h}(\mathbf{x}_k, \mathbf{y}_k) = \left[h(\mathbf{x}_k, y_{k,1}), \; \ldots, h(\mathbf{x}_k, y_{k,n_k})\right]^\top, \tag{15a}$$

$$\mathbf{y}_k = \left[y_{k,1}^\top, \; \ldots, \; y_{k,n_k}^\top\right]^\top, \tag{15b}$$

$$\mathbf{e}_k = \left[e_{k,1}^\top, \; \ldots, \; e_{k,n_k}^\top\right]^\top, \tag{15c}$$

$$R_k = \text{blkdiag}(R_{k,1}, \ldots, R_{k,n_k}), \tag{15d}$$

The given covariance matrix R is in block diagonal form, which follows from the assumption that the measurement noise terms are mutually independent.

In consequence, we obtain the following state-space model by defining a Gaussian prior density for the state vector,

$$\begin{aligned} \mathbf{x}_{k+1} &= F\mathbf{x}_k + \mathbf{w}_k, & \mathbf{w}_k &\sim \mathcal{N}(\mathbf{0}, Q_k), \\ \mathbf{0} &= \mathbf{h}(\mathbf{x}_k, \mathbf{y}_k) + \mathbf{e}_k, & \mathbf{e}_k &\sim \mathcal{N}(\mathbf{0}, R_k), \\ \mathbf{x}_0 &\sim \mathcal{N}(\boldsymbol{\mu}_0, P_0). \end{aligned} \tag{16}$$

The EKF regards this model to compute the posterior distribution by processing the measurements collected at time k.

E. Disclosing the Object Extent

The proposed model relies on the potential function representation that inherently characterizes the underlying object extent. Converting this description to an explicit definition of the object's extent might be of particular interest for various applications. For example, a detailed extent estimate might prove invaluable in path planning for an autonomous vehicle aiming to avoid obstacles, or for efficient grasp planning of a robot arm.

Let us define a binary class indicator c_* for a query point $\mathbf{u}_* \in \mathbb{R}^2$ such that $c_* = 1$ denotes that $\mathbf{u}_*$ is contained within the object extent, i.e., $\mathbf{u}_* \in \mathcal{S}$; while $c_* = 0$ indicates that $\mathbf{u}_*$ is a free-space point. To compute the corresponding class probabilities conditioned on the measurements collected up to time k, we will basically regard the probability density of the potential function at $\mathbf{u}_*$ conditioned on the acquired measurements, i.e., $p(f_*|\mathcal{Y}_{1:k})$ where $f_* \triangleq f(\mathbf{u}_*)$. To this end, we can write

$$f_* = H_*^{\mathbf{f}}\mathbf{f}_k + e^{\mathbf{f}}, \quad e^{\mathbf{f}} \sim \mathcal{N}(\mathbf{0}, R^{\mathbf{f}}) \tag{17}$$

where $H_*^{\mathbf{f}} \triangleq H^{\mathbf{f}}(\mathbf{u}_*)$, and using the posterior of the potential function representation computed by the EKF, $p(\mathbf{f}_k|\mathcal{Y}_{1:k}) = \mathcal{N}(\mu_{k|k}^{\mathbf{f}}, P_{k|k}^{\mathbf{f}})$, we obtain

$$\begin{aligned} &p(f_*|\mathcal{Y}_{1:k}) = \mathcal{N}(\mu_*, \sigma_*^2), \\ &\quad \text{where } \mu_* = H_*^{\mathbf{f}}\mu_{k|k}^{\mathbf{f}}, \quad \sigma_*^2 = H_*^{\mathbf{f}}P_{k|k}^{\mathbf{f}}{H_*^{\mathbf{f}}}^\top + R^{\mathbf{f}} \ . \end{aligned} \tag{18}$$

Subsequently, we follow the standard approach to obtain the class probabilities by squashing the estimate through a response function, [31, Ch. 3]. For the response function, we utilize the following cumulative density function (cdf) of the standard normal distribution, which in turn leads to a *so-called* probabilistic least-squares classifier [31, Ch. 6.5.1].

$$p\left(c_* = 1 \mid \mathcal{Y}_{1:k}\right) = \Phi\left(\frac{\alpha\mu_* + \beta}{\sqrt{1+\alpha^2\sigma_*^2}}\right) \tag{19}$$

Φ is the cdf of the standard normal distribution; and it is used as a sigmoid function with parameters α and β. Accordingly, we can calculate the other class' probability as $p\left(c_* = 0 \mid \mathcal{Y}_{1:k}\right) = 1 - p\left(c_* = 1 \mid \mathcal{Y}_{1:k}\right)$. Finally, after the class probabilities are computed, it is possible to recognize the query point as one of the classes {extent, free-space, unknown} by simply specifying some threshold values.

III. Simulation Results

In this section, we demonstrate the performance of the proposed method via simulation experiments. We investigate three different scenarios with different motion patterns and sensor characteristics.

In the first experiment, the object of interest is static throughout the scenario. At each time instant, we uniformly sample the test environment and collect 20 measurements from the surface of the object and 50 measurements from the surrounding free-space region. The detection points of the measurements are corrupted by i.i.d. Gaussian noise with covariance $0.03^2 I_2\ m^2$. The sampling time is used as $T = 0.1\ s$, and the total duration of the experiment is $10\ s$. We examine three different types of extents, i.e., a C-shape, a T-shape, and another one containing disconnected subparts. The illustrative results obtained at four time instants are shown in Fig. 4.

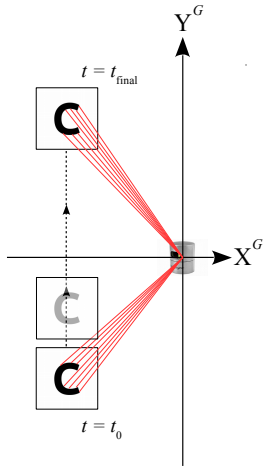


Fig. 2: Illustration of the second simulation experiment, where the object moves in a straight line at a constant speed. The measurements are simulated regarding a basic lidar model, taking into account how the sensor and the object are positioned relative to each other.

In the second experiment, we investigate an object moving along a linear path at a constant speed of $1\ m/s$, which is shown in Fig. 2. In this case, the observations are simulated to be acquired by a two-dimensional laser range scanner. The sampling time is used as $T = 0.1\ s$, the total duration of the experiment is $10\ s$. At each instant, the sensor emits 200 laser beams and reports the detection points of the ones reflected from the contour of the object. Additionally, for each laser beam, we produce 3 free-space measurements by randomly sampling the free section of the beam. All measurements are perturbed by i.i.d. Gaussian noise with covariance $0.01^2 I_2\ m^2$. The results are exhibited in Fig. 5.

In the third experiment, we consider an object performing rotational motion as depicted in Fig. 3. The measurements are simulated to be generated by a two-dimensional laser range scanner, whose characteristics are identical to the one utilized in the second experiment. The results are visualized in Fig. 6.

For all experiments, the tracker maintains the extent representation over 1024 basis points, which are equidistantly located on a grid over $[-3, 3]\ m \times [-3, 3]\ m$. We use a squared-exponential covariance function for the GP, $k\left(\mathbf{u}, \mathbf{u}'\right) = \sigma_f^2 \mathrm{e}^{-\frac{\left(\mathbf{u}-\mathbf{u}'\right)^2}{2l^2}}$, where the prior variance, σ_f^2, and the length-scale, l, are set to be 1 and 0.2, respectively. The prior mean of the kinematic variables are initialized at the ground truth, while the prior covariance is set to be $\bar{P}_0 = 1\mathrm{e}{-5} I_6$. The prior distribution of the extent variables are directly obtained from the used GP model.

In all cases, the proposed method satisfactorily estimates

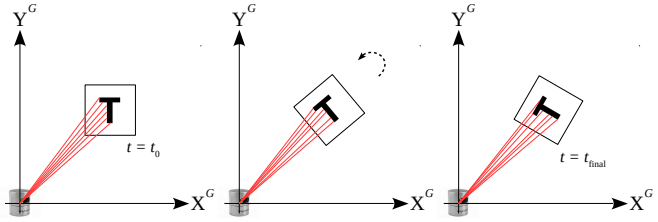


Fig. 3: Illustration of the third simulation experiment, in which the object rotates around a fixed point. The measurements are simulated regarding a basic lidar model, considering how the sensor and the object are positioned relative to each other.

the kinematic variables of the objects and learns the unknown potential function. In the first experiment, the information provided by the uniform measurements is successfully incorporated, hence at the end of the scenario, the uncertainty over the complete region is appropriately reduced, and the latent extent can easily be distinguished from the estimated potential function values. In the second experiment, due to the characteristics of the sensor, some sections of the object are always self-occluded in accordance with the object-sensor geometry. As the object moves along the trajectory, some of the previously unseen parts are explored by the laser beams, and the algorithm properly decreases the uncertainty on the observed portion while a larger uncertainty is associated with the unobserved regions. Notice that as the measurements are exclusively generated by the object contour in this case, the interior points of the extent remain mostly uncertain at the end of the experiment. Furthermore, the third experiment shows that the developed algorithm can successfully estimate the varying orientation of the object. As the object rotates, the method learns the underlying non-convex shape and appropriately decreases the uncertainty of the observed regions. All simulations are conducted in MATLAB 2021b on a standard laptop with an Intel Core i7 2.50 GHz CPU using 32 GB of RAM. The average computation time for a single time step of the tracker is measured to be approximately 90 ms.

IV. Conclusion and Future Work

Existing algorithms in the GPEOT framework assume that the object of interest is star-convex. While this assumption does not introduce a severe restriction and leads to sufficiently flexible methods for many tracking applications, we hereby focus on further extending the representation capabilities of the existing approaches. In this pursuit, we suggest describing the unknown extent by a potential function, which is modeled by a GP. Then, a state-space model consisting of both the kinematic variables and an approximate description of the underlying GP model is developed. The resulting tracker enables tracking arbitrarily shaped objects while learning their latent extent. We illustrate the performance of the proposed algorithm through simulation experiments.

References

[1] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, 2001.

[2] J. W. Koch, "Bayesian approach to extended object and cluster tracking using random matrices," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 3, pp. 1042–1059, Jul. 2008.

[3] M. Feldmann, D. Franken, and W. Koch, "Tracking of extended objects and group targets using random matrices," *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1409–1420, Apr. 2011.

[4] M. Baum and U. D. Hanebeck, "Extended object tracking with random hypersurface models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 1, pp. 149–159, Jan. 2014.

[5] C. Lundquist, U. Orguner, and F. Gustafsson, "Extended target tracking using polynomials with applications to road-map estimation," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 15–26, Jan. 2010.

[6] J. Lan and X. R. Li, "Tracking of extended object or target group using random matrix—Part II: Irregular object," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2012.

[7] K. Granström, C. Lundquist, and U. Orguner, "Tracking rectangular and elliptical extended targets using laser measurements," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2011.

[8] L. Hammarstrand, L. Svensson, F. Sandblom, and J. Sorstedt, "Extended object tracking using a radar resolution model," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 3, pp. 2371–2386, Jul. 2012.

[9] A. Zea, F. Faion, M. Baum, and U. D. Hanebeck, "Level-set random hypersurface models for tracking nonconvex extended objects," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 6, pp. 2990–3007, Dec. 2016.

[10] W. Cao, J. Lan, and X. R. Li, "Extended object tracking and classification using radar and ESM sensor data," *IEEE Signal Process. Lett.*, vol. 25, no. 1, pp. 90–94, Jan. 2018.

[11] H. Kaulbersch, J. Honer, and M. Baum, "A cartesian B-spline vehicle model for extended object tracking," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2018.

[12] J. Lan and X. R. Li, "Extended-object or group-target tracking using random matrix with nonlinear measurements," *IEEE Trans. Signal Process.*, vol. 67, no. 19, pp. 5130–5142, Oct. 2019.

[13] T. Baur, J. Reuter, A. Zea, and U. D. Hanebeck, "Shape tracking using Fourier-Chebyshev double series for 3D distance measurements," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2023.

[14] K. Granstrom, M. Baum, and S. Reuter, "Extended object tracking: Introduction, overview and applications," *J. Adv. Inf. Fusion*, vol. 12, no. 2, pp. 139–174, Dec. 2017.

[15] L. Mihaylova, A. Y. Carmi, F. Septier, A. Gning, S. K. Pang, and S. Godsill, "Overview of Bayesian sequential Monte Carlo methods for group and extended object tracking," *Digital Signal Process.*, vol. 25, pp. 1–16, Feb. 2014.

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

[17] T. Hirscher, A. Scheel, S. Reuter, and K. Dietmayer, "Multiple extended object tracking using Gaussian processes," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2016.

[18] B. Tuncer, M. Kumru, and E. Özkan, "Extended target tracking and classification using neural networks," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2019.

[19] Y. Guo, Y. Li, A. Xue, R. Tharmarasa, and T. Kirubarajan, "Simultaneous tracking of a maneuvering ship and its wake using Gaussian processes," *Signal Process.*, vol. 172, Jul. 2020.

[20] S. Lee and J. McBride, "Extended object tracking via positive and negative information fusion," *IEEE Trans. Signal Process.*, vol. 67, no. 7, pp. 1812–1823, Feb. 2019.

[21] K. Thormann, M. Baum, and J. Honer, "Extended target tracking using Gaussian processes with high-resolution automotive radar," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2018.

[22] B. Tuncer, M. Kumru, E. Özkan, and A. A. Alatan, "Extended object tracking and shape classification," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2018.

[23] M. Kumru and E. Özkan, "3D extended object tracking using recursive Gaussian processes," in *Proc. Int. Conf. Inf. Fusion (FUSION)*, 2018.

[24] ——, "Three-dimensional extended object tracking and shape learning using Gaussian processes," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 5, pp. 2795–2814, Mar. 2021.

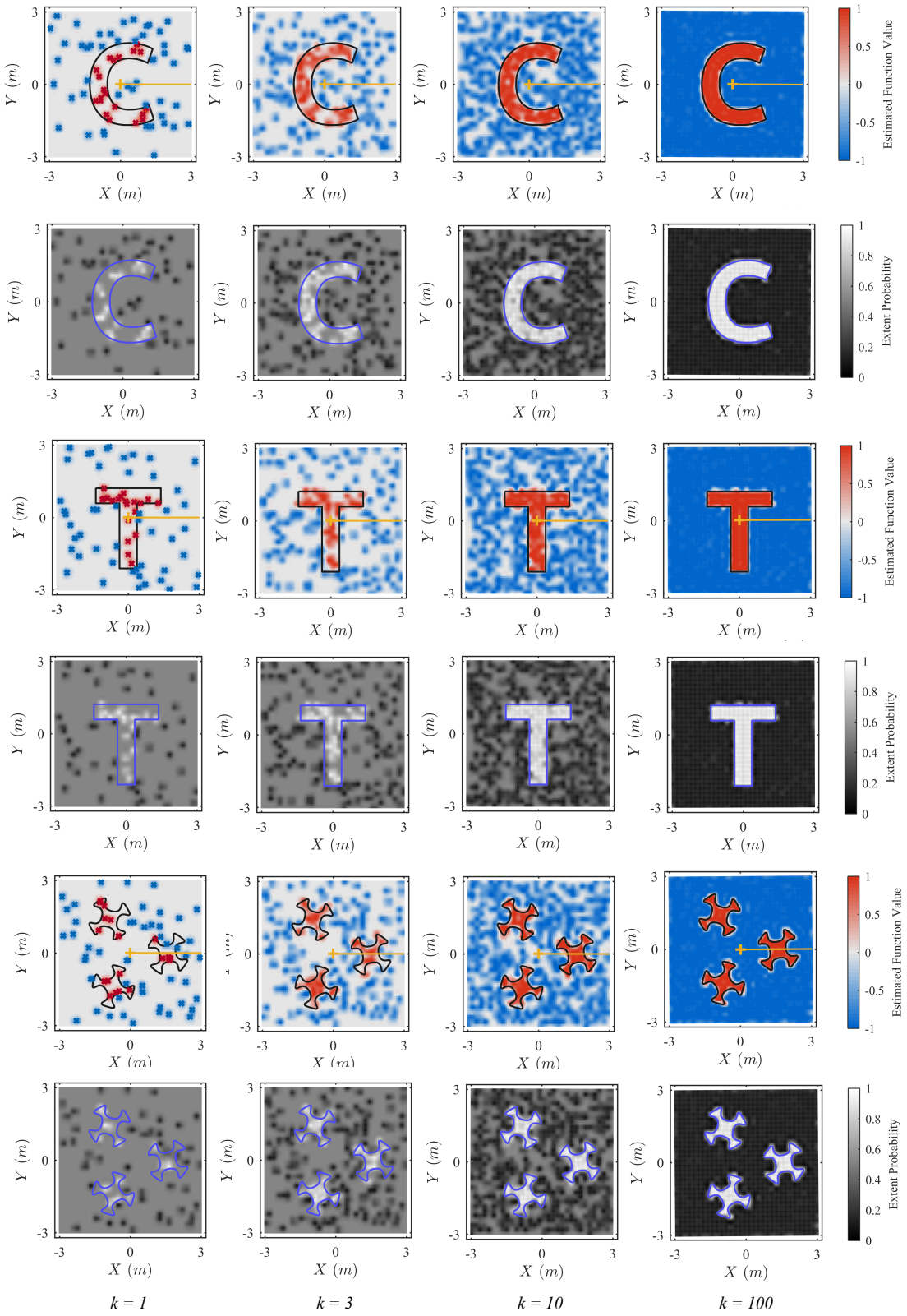


Fig. 4: Results obtained at some representative time instants $k = \{1, 3, 10, 100\}$ during the first simulation experiment. The true extent of the object is visualized by the solid curve. The estimated center and the orientation are indicated by the yellow plus sign and the straight line, respectively. For the first instant, measurements originated from the extent and the free space are shown by red and blue cross signs, respectively.

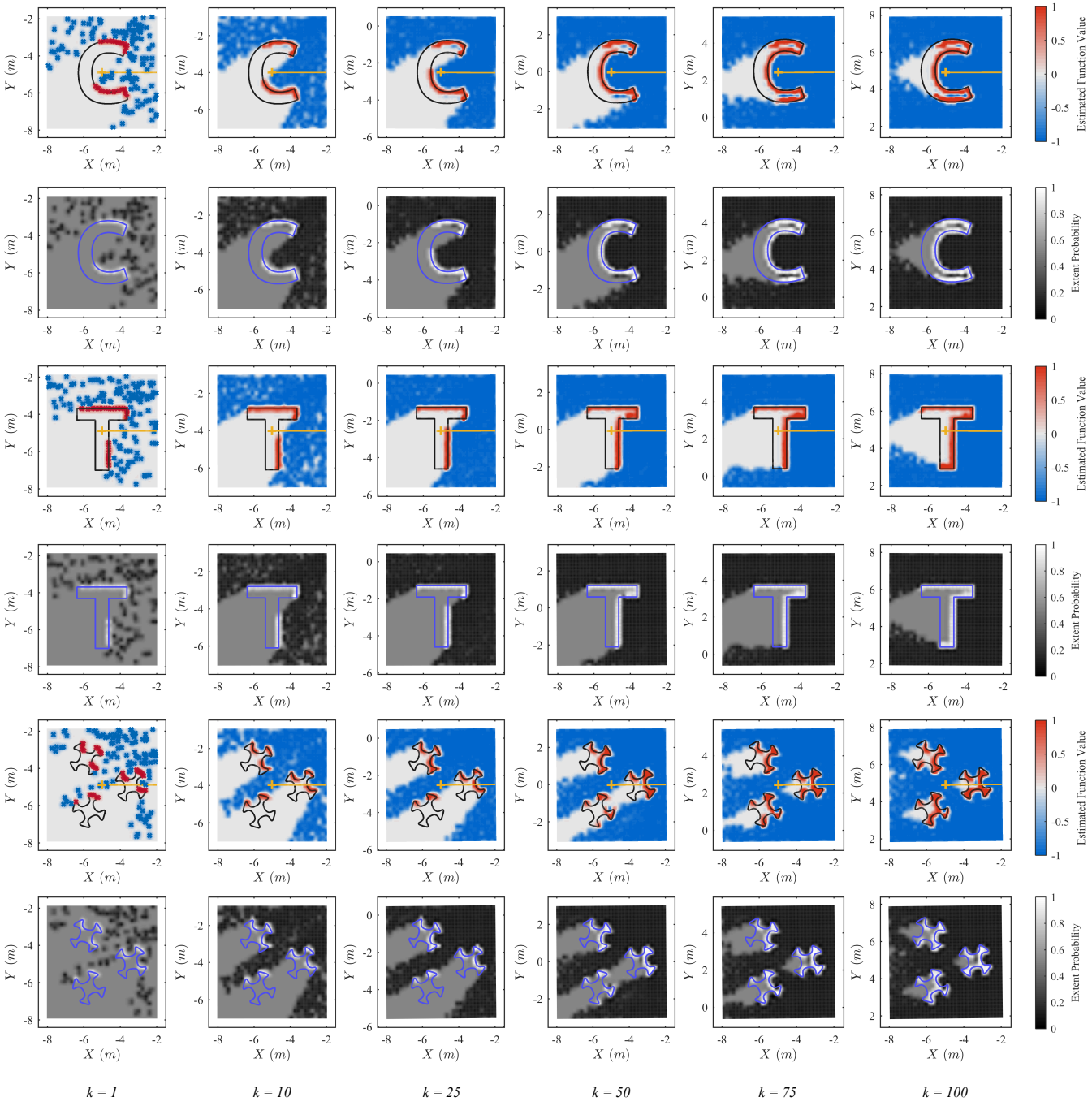
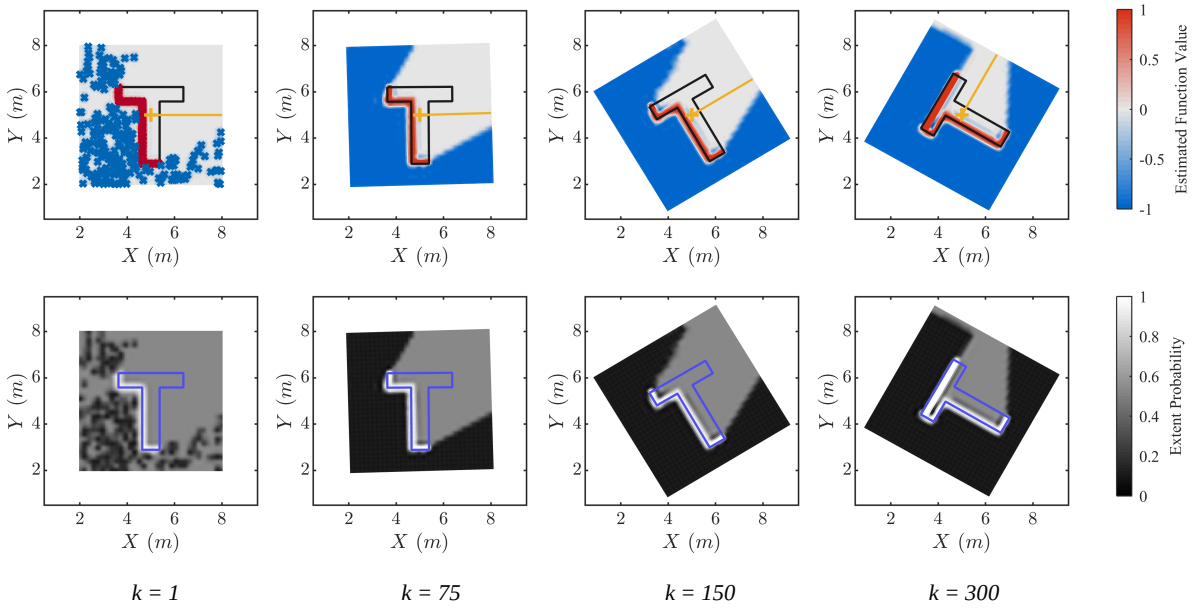
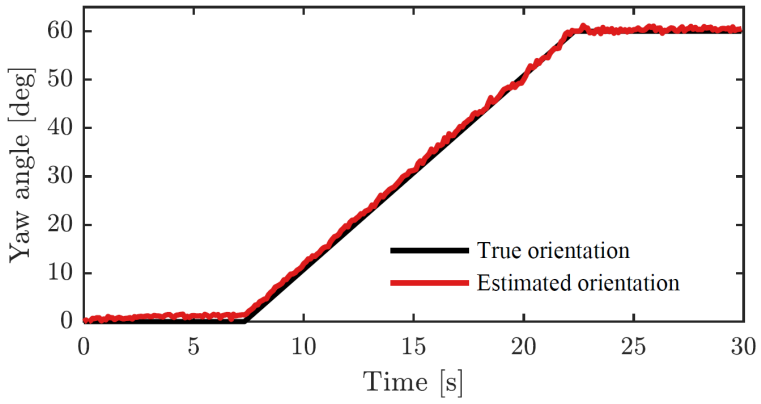


Fig. 5: Results obtained at some representative time instants $k = \{1, 10, 25, 50, 75, 100\}$ during the second simulation experiment. The true extent of the object is visualized by the solid curve. The estimated center and the orientation are indicated by the yellow plus sign and the straight line, respectively. For the first instant, measurements originated from the extent and the free space are shown by red and blue cross signs, respectively.



(a)



(b)

Fig. 6: Results obtained during the third simulation experiment. In Fig. (a) the outputs of the tracker are illustrated at some representative time instants $k = \{1, 75, 150, 300\}$. The true extent of the object is visualized by the solid curve. The estimated center and the orientation are indicated by the yellow plus sign and the straight line, respectively. For the first instant, measurements originated from the extent and the free space are shown by red and blue cross signs, respectively. In Fig. (b) the true and the estimated orientation are depicted.

[25] M. Kumru, H. Köksal, and E. Özkan, "Variational measurement update for extended object tracking using Gaussian processes," *IEEE Signal Process. Lett.*, vol. 28, pp. 538–542, Feb. 2021.

[26] D. Yang, Y. Guo, T. Yin, and B. Lin, "Cost-effective Gaussian processes based extended target tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 6, pp. 8282–8296, Dec. 2023.

[27] S. T. O'Callaghan and F. T. Ramos, "Gaussian process occupancy maps," *Int. J. Robot. Res.*, vol. 31, no. 1, pp. 42–62, Jan. 2012.

[28] M. G. Jadidi, J. V. Miro, and G. Dissanayake, "Gaussian processes autonomous mapping and exploration for range-sensing mobile robots," *Autonomous Robots*, vol. 42, no. 2, pp. 273–290, Feb. 2018.

[29] F. Ramos and L. Ott, "Hilbert maps: Scalable continuous occupancy mapping with stochastic gradient descent," *Int. J. Robot. Res.*, vol. 35, no. 14, pp. 1717–1730, 2016.

[30] E. Özkan, V. Šmídl, S. Saha, C. Lundquist, and F. Gustafsson, "Marginalized adaptive particle filtering for nonlinear models with unknown time-varying noise parameters," *Automatica*, vol. 49, no. 6, pp. 1566–1575, Jun. 2013.

[31] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.