A Gaussian Process Regression Approach for Point Target Tracking

Waqas Aftab, Lyudmila Mihaylova

Department of Automatic Control and Systems Engineering, University of Sheffield, S1 3JD, UK
{waftab1,l.s.mihaylova}@sheffield.ac.uk

***Abstract*—Target tracking performance relies on the match between the tracker motion model and the unknown target dynamics. The performance of these model-based trackers degrades when there is a mismatch between the model and the target motion. In this paper, a Gaussian process based approach, namely, Gaussian process motion tracker (GPMT) is proposed. The Gaussian process framework is flexible and can represent an infinite number of motion modes. The evaluation of the proposed approach is performed on challenging scenarios and is compared with popular single and multiple-model based approaches. The results show high accuracy of the predicted and estimated target position and velocity over challenging maneuver scenarios.**

***Index Terms*—Target Tracking, Gaussian Process, Motion Models.**

I. Introduction

Mulitple target tracking (MTT) [1], [2] aims at finding trajectories of multiple targets from sensor data and provides solutions to challenging data association problems. The MTT methods are broadly categorized to point target tracking (PTT) methods and extended / group target tracking (ETT / GTT) methods [3]–[6]. PTT deals with the target kinematics estimation, whereas ETT / GTT involves estimation of the target kinematics and the extent. PTT methods have been used in numerous systems and in diverse fields such as for air, land and sea traffic management, oceanography [7], medicine. Similarly, ETT / GTT has also been applied in various fields such as autonomous vehicles, urban management, environmental studies.

MTT methods provide a solution to two closely related problems, namely the data association and the filtering problem. The data association requires finding the correct measurement to trajectory assignment. Most filtering methods are model based and the choice of the motion and measurement models are of primary importance. This, in turn, also affects the data association process in the subsequent step. Various motion models have been proposed for tracking simple and complex target dynamics [8]. All of these models are equally applicable to the kinematics estimation required in the PTT, ETT and GTT.

A wealth of model-based filtering approaches [5], [6], [9] have been typically proposed. In most applications, the actual target dynamics cannot be represented exactly by the model. A modeling noise is added to capture this mismatch. Moreover, as the target motion is a continually changing process, a large number of models and model switchings are required to mimic the target motion and to determine good estimates. Even if the switching process is known a priori, it is not efficient to design a practical system with a large number of models. The modeling noise is again included to reflect inaccuracies of the motion representation. The state estimation using model switching among a large or even an infinite number of models can be achieved using the model-free approach proposed in this paper.

A wealth of research has been done on time series prediction and estimation using Gaussian process (GP) methods such as [10], [11]. However, the GP approach has not been widely studied by the target tracking community for estimating the target motion, especially for highly maneuvering targets. A GP based trajectory estimation has been proposed for simultaneous localization and mapping [12], [13], where the target kinematics are not considered as highly maneuverable. A novel approach for extended target tracking using GPs has been proposed in [14]. In this work and the other similar works [15]–[17], the target shape estimation is proposed using a GP model whereas the target (center/average) motion is filtered using the model based approaches proposed for PTT. An overlapping mixture of GPs (OMGP) has also been proposed in [18] to solve the data association problem arising in the MTT. To the best of our knowledge, all previous approaches for the estimation of the target kinematics are model based. Hence, for the first time, this paper proposes a data-driven approach for the target kinematics estimation. A Gaussian process motion tracker (GPMT) is proposed in this paper and it represents the target kinematics as a GP regression [10].

A Gaussian process motion tracker (GPMT), proposed in this paper, is a data driven approach based on the GP regression [10]. The GP is a stochastic process that uses a prior distribution over functions and training data to predict the functional values at points not included in the training data. The GPMT models the target motion as a nonlinear function of time using a GP prior over this unknown function. The mean of the GP represents the mean of the function matched to the target dynamics. Since the GP is a distribution over functions, an infinite number of functions or models selection can be achieved using the GPMT. The GPMT uses the available data to select the model and then find the current or predicted estimate based on the chosen model. A GP, being a batch regression approach, is not suitable for real-time temporal systems. The GPMT assumes that the model selection depends on the training data in the near past, only. Hence, the batch regression problem is reduced and the real-time implementation is achieved.All MTT model-based approaches rely on the calculation of current and predicted state estimates. In contrast, the GPMT does not require such a two-step process. The estimation and the prediction processes run independently of each other. Hence, when only estimates at the current time moment are required, the prediction process can be omitted.

The GPMT based filter gives the location estimates. In most applications, the consideration of higher order time derivatives of the location is also important. This can be achieved by using the derivative of the GPs [10]. A first order time derivative extension of the GPMT is presented in this paper for estimating the target velocity. The same concept can be extended to determine other higher order derivatives.

The rest of the paper is paper is structured as follows. Sections II-A and II-B provide a brief overview of the state space models used in MTT and the motion models. The theoretical background of the GP and the derivative GP is covered in Sections II-C and II-D, respectively. The proposed model is explained in Section III and the extension of the proposed model is given in Section III-B. The performance evaluation is done in Section IV followed by conclusions.

II. Background Knowledge

A. State Space Model for Multiple Target Tracking

This section gives a brief overview of the state space model used in MTT. The MTT algorithms deal with data association and maintain the trajectories of multiple targets by an appropriately chosen estimation approach. The assignment of measurements to respective target trajectories is key for achieving accurate results. The following state-space model represents the target dynamics and the sensor model:

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{w}_k), \tag{1}$$

$$\boldsymbol{z}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{v}_k), \tag{2}$$

where $\boldsymbol{x}$ and $\boldsymbol{z}$ represent, respectively, the state and the measurement vector, k is the discrete time index, $\boldsymbol{f}$ and $\boldsymbol{h}$ represent the state motion model and the measurement function, respectively, and $\boldsymbol{w}$ and $\boldsymbol{v}$ are the process and measurement noise vectors, respectively.

B. Overview of Target Motion Models and Filtering Methods

This section gives a brief overview of target motion models. A comprehensive survey of the motion models can be found in the three survey papers [8], [19], [20]. The models have been categorized and studied as non multiple-model (non-MM) [8], decision-based methods [19] and multiple-model methods (MM) [20]. Their practical implementation has been discussed extensively in [1].

The non-MM motion models can be categorized from simple to complex models based on the assumption made on the coordinate coupling and the temporal correlations as shown in Fig. 1. The simplest models assume that the target motion is uncoupled across coordinates and uncorrelated in time. The acceleration and jerk[1] models are assumed to be a white noise process in nearly constant velocity (NCV) [1] and nearly constant acceleration (NCA) [21] models, respectively. The velocity and acceleration are assumed almost constant in the NCV and NCA, respectively. The $(n+1)$-th order polynomial model [21] is achieved by assuming the $(n+1)$-th position derivative to be white noise and an almost constant n-th position derivative. These models assume zero coupling among the coordinates. This assumption is relaxed in the nearly coordinated turn (NCT) [1], [21] based models, which in addition to the coupling assume an almost constant forward speed and turn rate. Such models provide better estimates during target maneuvers. The performance is degraded, in comparison to the NCV and NCA, when the target follows a linear motion. Since the true target dynamics is a continuous process, the motion parameters are correlated in time. The time correlation is considered in relatively complex models such as Singer [22] and jerk models [23]. The coordinate coupled version of the Singer model is also proposed as n-th order Markov model [24]. The target acceleration based models provide better estimates in the presence of both the position and the velocity measurements [1]. These models are applicable to the two dimensional (2D) tracking case directly and are extended to the three dimensional (3D) case as well. Often the Kalman filter [25] is applied when there is no measurement data association filter.

Two other groups of methods - the decision-based and the MM methods [19], [20], use a bank of filters to provide the estimates. The decision-based methods choose a single filter to give the state estimates at any given time. On the other hand, the MM filters combine the filters to give the state estimates. These have been classified into three generations [20],the first generation [26], [27] methods, the second generation namely interacting MM (IMM)) [2], [28]) and the third generation referred to as the variable structure IMM (VSIMM) [29]. Considering the computational complexity and the performance, the interacting multiple-model (IMM) has been shown to be the most cost-effective method. The IMM, also called Fixed Grid IMM (FGIMM), has been successfully applied in various tracking systems [1]

C. Gaussian Process

A GP is a stochastic process, any finite realization of which is jointly Gaussian distributed. The GP, defined by a mean function and a covariance kernel, has been used for solving problems involving regression and classification [10]. It allows a non-parametric functional mapping from the input to the output space. These spaces can be single, multiple or a combination of them. As a non-parametric method, the GP method relies on training data for the output prediction at the unknown input locations. The GP mean and the covariance kernel act as a prior on the prediction process. In most applications, the mean is kept

[1]The time derivative of acceleration is called a jerk.

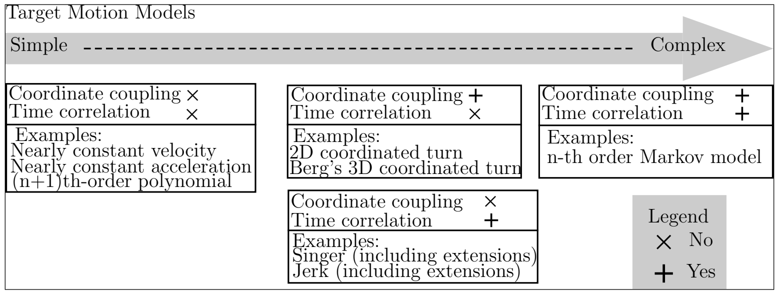


Fig. 1. **Classification of motion models for target tracking**.This figure shows the classification of motion models based on the assumptions of the coordinate coupling and time correlations. The '+' and 'x' represent, respectively, the presence and absence of the model assumption e.g. the left most box represents models that assume no coordinate coupling or time correlations. The model complexity increases from left to right.

constant or zero, which means that there is no prior information regarding the mean behavior of the modeled function. Hence, the covariance kernel is the most important design parameter of the GP based models. The details of different mean and covariance kernels can be found in [10].

Suppose, a GP is used to model a nonlinear function f, which relates the one dimensional input r to the one dimensional output s as given below:

$$s = f(r), \quad f \sim GP(\mu, k(r, r')), \tag{3}$$

where μ and k represent, respectively, the mean and the covariance kernel of the GP. Suppose, the output is known at n input locations. The output at unknown $n^\star$ input locations can be determined using the GP as given below:

$$\mathbb{E}[\boldsymbol{s}^\star] = \boldsymbol{\mu} + \boldsymbol{K}_{\boldsymbol{r}^\star \boldsymbol{r}} (\boldsymbol{K}_{\boldsymbol{rr}} + \sigma^2 \boldsymbol{I}_n)^{-1} (\boldsymbol{s} - \boldsymbol{\mu}), \tag{4}$$

$$cov[\boldsymbol{s}^\star] = \boldsymbol{K}_{\boldsymbol{r}^\star \boldsymbol{r}^\star} - \boldsymbol{K}_{\boldsymbol{r}^\star \boldsymbol{r}} (\boldsymbol{K}_{\boldsymbol{rr}} + \sigma^2 \boldsymbol{I}_n)^{-1} \boldsymbol{K}_{\boldsymbol{r}\boldsymbol{r}^\star}, \tag{5}$$

where $\boldsymbol{s}^\star$ and $\boldsymbol{r}^\star$ represent, respectively, the output and input vectors at $n^\star$ locations, $\mathbb{E}[\cdot]$ is the mathematical expectation operator, $\mathbb{E}[\boldsymbol{s}^\star]$ and $cov[\boldsymbol{s}^\star]$ represent, respectively, the predicted mean and the predicted covariance of the GP, $\boldsymbol{r}$ and $\boldsymbol{s}$ are the n-dimensional input and output training data vectors, respectively, $\boldsymbol{\mu}$ represents the n-dimensional GP mean vector at the training input locations, σ^2 is the noise variance, $\boldsymbol{K}$ denotes the GP covariance matrix, $\cdot^{-1}$ denotes the inverse matrix and $\boldsymbol{I}_n$ represents an n-dimensional identity matrix. A GP covariance matrix between a j-dimensional vector $\boldsymbol{p}$ and an l-dimensional vector $\boldsymbol{q}$ is given below:

$$\boldsymbol{K}_{\boldsymbol{pq}} = \begin{bmatrix} m(p_1, q_1) & m(p_1, q_2) & \cdots & m(p_1, q_l) \\ m(p_2, q_1) & m(p_2, q_2) & \cdots & m(p_2, q_l) \\ \vdots & \vdots & \ddots & \vdots \\ m(p_j, q_1) & m(p_j, q_2) & \cdots & m(p_j, q_l) \end{bmatrix}. \tag{6}$$

The parameters of the GP mean function and the covariance kernel are called *hyperparameters*. The hyperparameters can also be found, e.g. by maximizing the likelihood ($\frac{-1}{2} \boldsymbol{s}' \boldsymbol{K}_{\boldsymbol{rr}}^{-1} \boldsymbol{s} - \frac{1}{2} \log |\boldsymbol{K}_{\boldsymbol{rr}}| - \frac{n}{2} \log 2\pi$). The process of finding (*learning*) the hyperparameters is a non-convex optimization problem which is computationally complex and might sometimes be locked in local minima. For some applications, hyperparameters can be learned or set to a fixed value based upon prior knowledge. This, in turn, improves the processing time and avoids the prediction based on poor optimization results.

D. Derivative Gaussian Process

The derivative of a GP is also a GP [10]. The GP regression using derivative measurements has been proposed in [30]. In typical radar tracking applications, these derivative measurements are not available. The GP inference of a function and its derivatives using the observations is proposed in [31].

In what follows we present the new approach, using the observations only. It requires the GP covariance kernel to be as many times differentiable as the order of the desired derivative process. Next, we describe a second order derivative GP which is in the heart of the developed approach.

Consider the GP model (3), with mean $\mu = 0$. The joint probability density function of the known output vector $\boldsymbol{s}$, the unknown output vector $\boldsymbol{s}^\star$ and its higher order derivatives $\boldsymbol{s'}^\star = \frac{d\boldsymbol{s}^\star}{dr}$ and $\boldsymbol{s''}^\star = \frac{d^2\boldsymbol{s}^\star}{dr^2}$ is also Gaussian. The means and the covariance matrices of the first- and second-order derivative processes are given below:

$$\mathbb{E}\begin{bmatrix} \boldsymbol{s'}^\star \\ \boldsymbol{s''}^\star \end{bmatrix} = \begin{pmatrix} \frac{\partial \boldsymbol{K}^\star}{\partial \boldsymbol{r}^\star} \\ \frac{\partial^2 \boldsymbol{K}^\star}{\partial \boldsymbol{r}^{\star 2}} \end{pmatrix} \left[\boldsymbol{K} + \sigma^2 \boldsymbol{I}_n\right]^{-1} \boldsymbol{s}, \tag{7}$$

$$cov\left[\boldsymbol{s'}^\star\right] = \frac{\partial^2 \boldsymbol{K}^{\star\star}}{\partial \boldsymbol{r}^\star \partial \boldsymbol{r}} - \frac{\partial \boldsymbol{K}^\star}{\partial \boldsymbol{r}^\star} [\boldsymbol{K} + \sigma^2 \boldsymbol{I}_n]^{-1} \left[\frac{\partial \boldsymbol{K}^\star}{\partial \boldsymbol{r}^\star}\right]^T, \tag{8}$$

$$cov\left[\boldsymbol{s''}^\star\right] = \frac{\partial^4 \boldsymbol{K}^{\star\star}}{\partial \boldsymbol{r}^{\star 2} \partial \boldsymbol{r}^2} - \frac{\partial^2 \boldsymbol{K}^\star}{\partial \boldsymbol{r}^{\star 2}} [\boldsymbol{K} + \sigma^2 \boldsymbol{I}_n]^{-1} \left[\frac{\partial^2 \boldsymbol{K}^\star}{\partial \boldsymbol{r}^{\star 2}}\right]^T, \tag{9}$$

where $\boldsymbol{K}^\star = \boldsymbol{K}_{\boldsymbol{r}^\star \boldsymbol{r}}$, $\boldsymbol{K} = \boldsymbol{K}_{\boldsymbol{rr}}$, $\boldsymbol{K}^{\star\star} = \boldsymbol{K}_{\boldsymbol{r}^\star \boldsymbol{r}^\star}$, ∂ denotes the partial derivative and $\cdot^T$ is the matrix transpose.

III. Gaussian Process Motion Tracker

The GPMT relies on past measurements to estimate the current and predict the future states. This model assumes that the coordinate coupling is weak enough to be ignored. This coupling can be included by extending the proposed approach with coupled GP [32]. The proposed model is in 2D and can be extended to 3D in a straightforward way.

A. 2D Gaussian Process Motion Tracker

The 2D GPMT assumes that the Cartesian x and y position coordinates of the target are not correlated in the last time instant, but these position coordinates could be correlated in past moments in time. The past measurements and the GP model, essentially the GP covariance kernel, are used to determine the nonlinear function. The unknown nonlinear functions mapped to the x and y Cartesian position coordinates of the target using a GP in the GPMT are given below:

$$x = f^x(t), \quad y = f^y(t), \tag{10}$$

$$f^x \sim GP(0, k(t,t')), \quad f^y \sim GP(0, k(t,t')), \tag{11}$$

where f^x and f^y are the corresponding non-linear latent functions and t is the (input) time domain parameter. In this paper we adopt the squared exponential covariance kernel [10] for the two GPs. Other kernels can also be explored depending upon the application.

The GPMT considers the d most recent measurement samples, also called depth of the tracker, for the state prediction and estimation as given below:

$$\alpha_{k+1|k} = \boldsymbol{K}_{\tilde{u}\boldsymbol{u}}[\boldsymbol{K}_{\boldsymbol{u}\boldsymbol{u}} + \sigma^2 \boldsymbol{I}_d]^{-1} \boldsymbol{z}_{\boldsymbol{u}}, \tag{12}$$

$$\rho^2_{k+1|k} = \boldsymbol{K}_{\tilde{u}\tilde{u}} - \boldsymbol{K}_{\tilde{u}\boldsymbol{u}}[\boldsymbol{K}_{\boldsymbol{u}\boldsymbol{u}} + \sigma^2 \boldsymbol{I}_d]^{-1} \boldsymbol{K}^T_{\tilde{u}\boldsymbol{u}}, \tag{13}$$

$$\alpha_{k+1|k+1} = \boldsymbol{K}_{\hat{u}\boldsymbol{u}'}[\boldsymbol{K}_{\boldsymbol{u}'\boldsymbol{u}'} + \sigma^2 \boldsymbol{I}_d]^{-1} \boldsymbol{z}_{\boldsymbol{u}'}, \tag{14}$$

$$\rho^2_{k+1|k+1} = \boldsymbol{K}_{\hat{u}\hat{u}} - \boldsymbol{K}_{\hat{u}\boldsymbol{u}'}[\boldsymbol{K}_{\boldsymbol{u}'\boldsymbol{u}'} + \sigma^2 \boldsymbol{I}_d]^{-1} \boldsymbol{K}^T_{\hat{u}\boldsymbol{u}'}, \tag{15}$$

where $\tilde{u} = k+1$, $\boldsymbol{u} = [k-d+1, k-d+2, \cdots, k]^T$, $\hat{u} = k$, $\boldsymbol{u'} = [k-d+2, k-d+3, \cdots, k+1]^T$, α and ρ^2 denote, respectively, the position means and the variances, the subscripts $k+1|k$ and $k+1|k+1$ represent, respectively, the predicted and the estimated states, σ^2 is the measurement noise variance and $\boldsymbol{z_a}$ represents the measurement vector consisting of samples corresponding to time vector $\boldsymbol{a}$. At each time-step the hyperparameters are learnt by maximizing the likelihood.

B. First Order 2D GPMT

The 2D GPMT, given in Section III-A, provides both the predicted and estimated position coordinates of the target. The first-order derivatives with respect to position coordinates can be determined, based on the the derivations from [30] and [31], using the first order 2D GPMT (FO-GPMT). These are given below:

$$\dot{\alpha}_{k+1|k} = \frac{\partial \boldsymbol{K}_{\hat{u}\boldsymbol{u}}}{\partial \hat{u}}[\boldsymbol{K}_{\boldsymbol{u}\boldsymbol{u}} + \sigma^2 \boldsymbol{I}_d]^{-1} \boldsymbol{z}_{\boldsymbol{u}}, \tag{16}$$

$$\dot{\rho}^2_{k+1|k} = \frac{\partial^2 \boldsymbol{K}_{\hat{u}\hat{u}}}{\partial \hat{u}\partial \hat{u}} - \frac{\partial \boldsymbol{K}_{\hat{u}\boldsymbol{u}}}{\partial \hat{u}}[\boldsymbol{K}_{\boldsymbol{u}\boldsymbol{u}} + \sigma^2 \boldsymbol{I}_d]^{-1}\left[\frac{\partial \boldsymbol{K}_{\hat{u}\boldsymbol{u}}}{\partial \hat{u}}\right]^T, \tag{17}$$

$$\dot{\alpha}_{k+1|k+1} = \frac{\partial \boldsymbol{K}_{\hat{u}\boldsymbol{u}'}}{\partial \hat{u}}[\boldsymbol{K}_{\boldsymbol{u}'\boldsymbol{u}'} + \sigma^2 \boldsymbol{I}_d]^{-1} \boldsymbol{z}_{\boldsymbol{u}'}, \tag{18}$$

$$\dot{\rho}^2_{k+1|k+1} = \frac{\partial^2 \boldsymbol{K}_{\hat{u}\hat{u}}}{\partial \hat{u}\partial \hat{u}} - \frac{\partial \boldsymbol{K}_{\hat{u}\boldsymbol{u}'}}{\partial \hat{u}}[\boldsymbol{K}_{\boldsymbol{u}'\boldsymbol{u}'} + \sigma^2 \boldsymbol{I}_d]^{-1}\left[\frac{\partial \boldsymbol{K}_{\hat{u}\boldsymbol{u}'}}{\partial \hat{u}}\right]^T, \tag{19}$$

where $\dot{\alpha}$ and $\dot{\rho}^2$ denote, respectively, the mean and variance of the first order derivatives with respect to position coordinates.

IV. Performance Validation

The GPMT performance is validated over five challenging maneuvering testing examples and over 10000 Monte Carlo independent runs. The proposed approach is compared with three other filters based on the NCV, FGIMM and Singer models [8]. The root mean square error (RMSE) of the position and the velocity are considered as the performance comparison parameters. A performance grade $\in \{1, 2, 3, 4\}$ is assigned to each model for each RMSE value of each scenario. The grade value of 1 is assigned to the best and 4 to the worst, out of the four methods. Testing scenarios in which the motion models used to generate the trajectory of the targets are the same with the models used in the tracking algorithms are called "matched". The five scenarios are explained below:

1) **S1: Uniform motion**. The target velocity is constant and the scenario matches the NCV and the FGIMM filters.
2) **S2: Coordinated turns matched**. The target motion is modeled using the NCT ($25\,\mathrm{deg}/s$ for 8s) and NCV motion models. This scenario is matched to FGIMM filter.
3) **S3: Coordinated turns mis-matched**. This is similar to S2 but the NCT ($12\,\mathrm{deg}/s$ for 20s) model is not matched to the FGIMM. The scenario is not matched to any filters.
4) **S4: Singer matched**. The target motion is modelled using a Singer acceleration model with maximum possible acceleration $A_{max} = 50m/s^2$, probability of no-acceleration $P_0 = 0.4$, probability of maximum acceleration $P_{max} = 0.1$ and maneuver time constant $\tau_m = 8s$.
5) **S5: Singer mis-matched**. This scenario is similar to S4 with following changes in the parameters, $A_{max} = 2m/s^2$, $P_0 = 0.6$ and $\tau_m = 25s$. This scenario is not matched to any filter.

The initial target velocity in each coordinate is chosen randomly between $150m/s$ and $250m/s$ and the target maintains uniform motion for initial $5s$ in all scenarios. The total duration is $100s$ and the measurement noise standard deviation is $\sigma = 25m$. The position and velocity RMSE are used to evaluate the performance. A sample trajectory of each scenario is shown in Fig. 2. A 15% initialization noise is added to all filters.

A. Filter Parameters

a) NCV-KF: The process noise variance is $500m^2/s^2$. The noise variance value is chosen high to prevent the filter from diverging during sharp maneuvers of scenarios S2, S3 and S4.

b) Singer-KF: The model parameters are chosen the same as in S4.

c) FGIMM: The fixed grid is modeled using a single NCV model and 2 NCT models. The rate of turns of the NCT models are set to $\{-25, 25\}\,\mathrm{deg}/s$. The Markov transition probability of the same mode is set to 0.7 and for changing the mode is

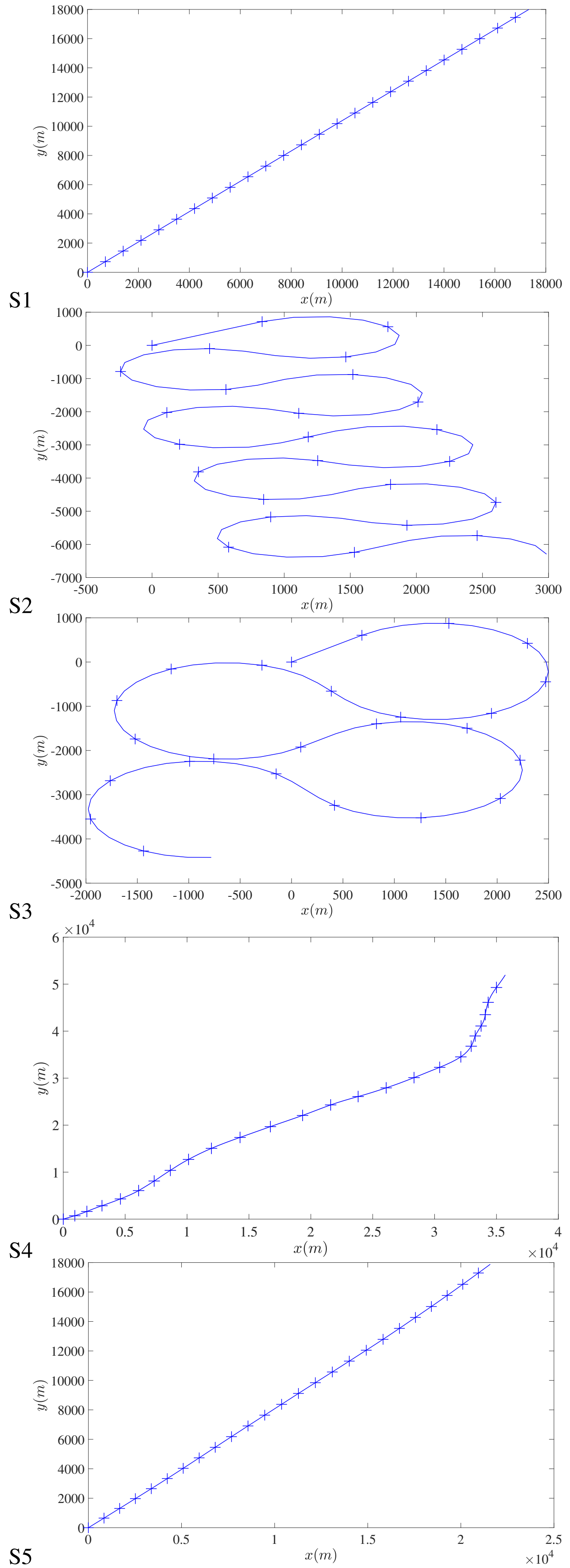


Fig. 2. **Sample trajectory**. The figure shows a sample trajectory for each scenario. The trajectory starts from origin (0,0) in all scenarios.

TABLE I
PERFORMANCE GRADES OF THE PROPOSED APPROACH

	Prediction Grades				**Estimation Grades**			
Scenario	x	y	v_x	v_y	x	y	v_x	v_y
S1	1	1	1	1	1	1	1	1
S2	2	2	2	2	2	2	2	2
S3	1	1	1	1	1	1	1	1
S4	2	2	2	2	2	2	2	2
S5	1	1	1	1	1	1	1	1

0.15, the initial model probability vector is $\{0.15, 0.7, 0.15\}$ and the process noise variance is set to $7.57 \times 10^{-8} m^2/s^2$ for each model.

d) FO-GPMT: The tracker depth is set to $d = 10$ samples.

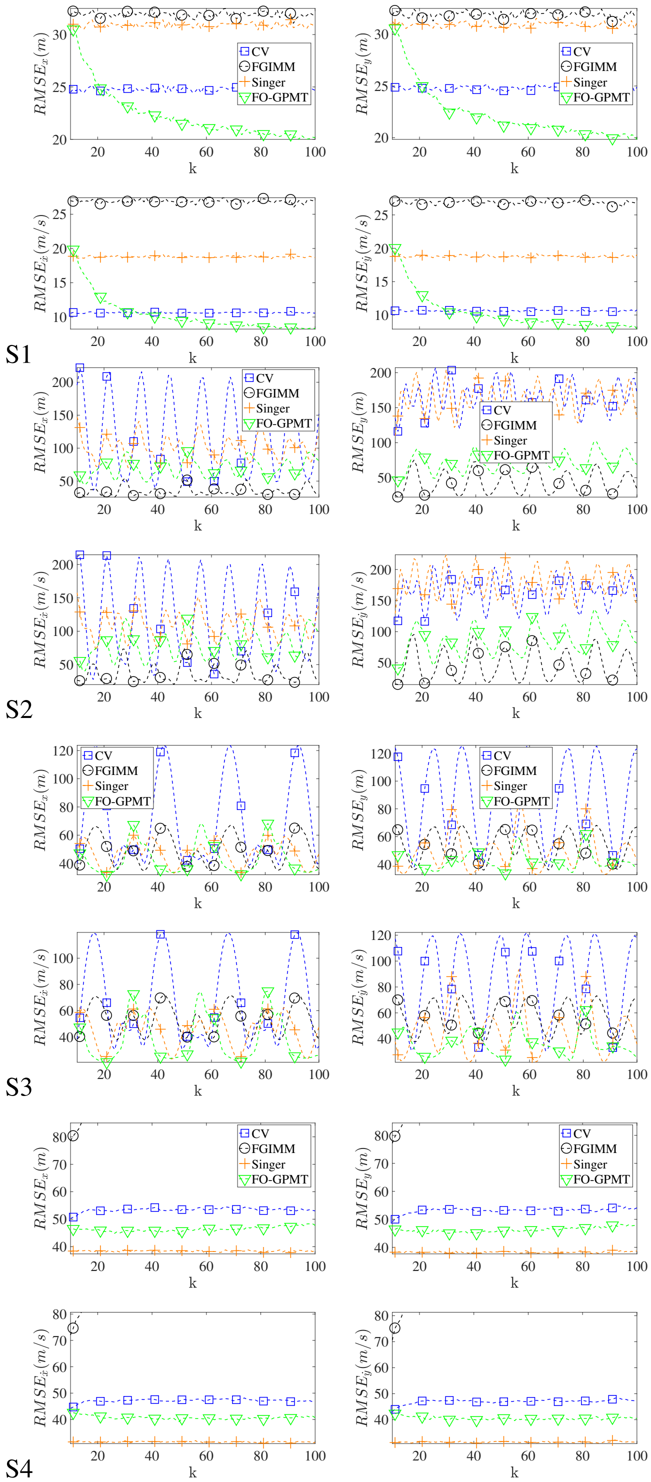
B. Results

The graphical and numerical comparisons of the prediction process are given in Figures 3 and 4, respectively. Similarly, the comparisons of the estimation process are given in the Figures 5 and 6. The performance grades of the proposed approach are given in Table I. It can be observed that both the prediction and estimation accuracies of the proposed approach are better than the model based approaches in the mismatched scenarios, S3 and S5. In matched scenarios, it is second best to the matched filter only. An improved performance in the estimates is observed for the Kalman filter based methods. This improvement in the model based approaches in the estimation process, as compared to their respective prediction process, can be attributed to the Kalman filter rather than to the performance of the underlying model.

The NCV and the FGIMM filters are expected to give grade 1 performance for S1. However, the proposed approach performs better than both matched filters. The FGIMM and Singer filters perform best for S2 and S4, respectively, as the simulated scenario is exactly matched to the motion model of the filter. The performance of the FGIMM is worse for mismatched scenarios and it even diverges for S4. The Singer based filter performs worst for the uniform motion based scenario that is S1. This motion is one of the common modes in many tracking applications. For example, the aerial targets tracking systems are expected to track airliners which are mostly moving under nearly constant motion.

V. CONCLUSIONS

This paper presents a GP model free approach for filtering and prediction of multiple target trajectories. The proposed GPMT approach can provide both estimates in the current time moment and future predictions of the target positions. A first-order extension of the proposed approach, FO-GPMT, is also proposed for the predicted and estimated velocity. The proposed approach does not require initialisation and the estimation and the prediction processes run independently. The evaluation of the proposed model-free approach is compared with model-based approaches. The results show that the FO-GPMT predicted and estimated positions are more accurate than the model-based results. This also provides an improvement to



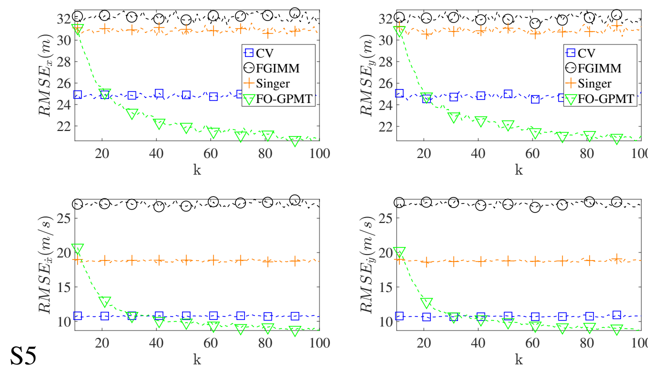


Fig. 3. **Prediction performance**. The figure shows the prediction performance results of 10000 Monte Carlo runs for the given five scenarios. Considering all the 5 scenarios, the proposed approach performs best due to its adaptability.

	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	25	25	11	11
2	Singer	31	31	19	19
3	FGIMM	32	32	27	27
4	FO-GPMT	22	22	10	10

S1

	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	115	159	113	160
2	Singer	98	155	107	171
3	FGIMM	35	43	37	47
4	FO-GPMT	70	72	81	86

S2

	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	69	86	64	82
2	Singer	46	49	44	47
3	FGIMM	49	54	53	57
4	FO-GPMT	43	43	38	36

S3

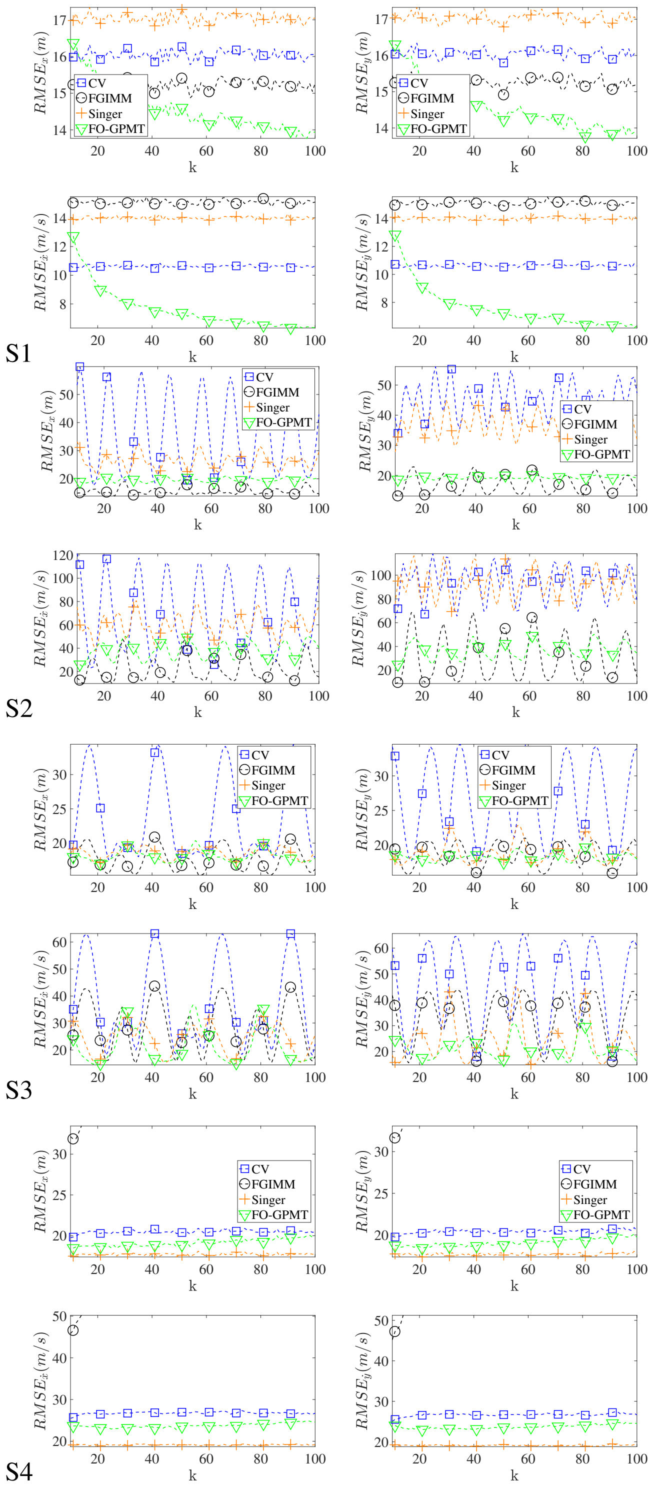
	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	52	52	46	46
2	Singer	38	38	31	31
3	FGIMM	NaN	NaN	NaN	NaN
4	FO-GPMT	46	46	41	41

S4

	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	25	25	11	11
2	Singer	31	31	19	19
3	FGIMM	32	32	27	27
4	FO-GPMT	23	23	11	11

S5

Fig. 4. **Predicted mean errors**. This figure shows 5 tables providing comparison of the RMSE values for the 5 scenarios. Each Table is tagged with the scenario label at the left bottom. The cell value of NaN means that the corresponding filter diverged.



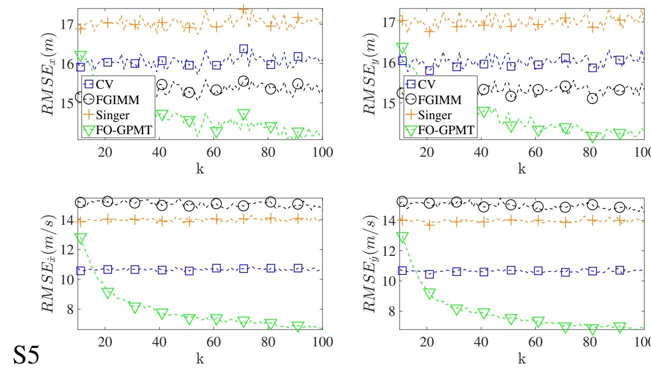


Fig. 5. **Estimation performance**. The figure shows the estimation performance results of 10000 Monte Carlo runs for the given five scenarios. It can be observed that the estimation performance of the model based approaches improves as compared to their respective prediction performance. This improvement is due to the Kalman filter, which improves the prediction performance using the innovation process.

	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	16	16	11	11
2	Singer	17	17	14	14
3	FGIMM	15	15	15	15
4	FO-GPMT	15	15	8	8

S1

	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	35	45	66	92
2	Singer	26	36	58	89
3	FGIMM	16	17	22	32
4	FO-GPMT	20	19	38	37

S2

	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	23	27	36	45
2	Singer	18	19	24	25
3	FGIMM	18	18	29	33
4	FO-GPMT	18	18	22	21

S3

	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	20	20	26	26
2	Singer	18	18	19	19
3	FGIMM	NaN	NaN	NaN	NaN
4	FO-GPMT	19	19	24	24

S4

	Model	rmse-x	rmse-y	rmse-vx	rmse-vy
1	CV	16	16	11	11
2	Singer	17	17	14	14
3	FGIMM	15	15	15	15
4	FO-GPMT	15	15	8	8

S5

Fig. 6. **Estimated mean errors**. This figure shows 5 tables providing comparison of the mean estimation errors for the 5 scenarios. Each Table is tagged with the scenario label at the bottom left. The cell value of NaN means the corresponding filter diverged.

the data association. Future work will focus on evaluation of the uncertainty propagation using different covariance kernels and theoretically.

Appendix A
Discrete time Kalman filter

The discrete time Kalman filter is based on the following discrete time system model:

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{F}_k \boldsymbol{x}_k + \boldsymbol{w}_k, \quad \boldsymbol{w}_k \sim \mathcal{N}(0, \boldsymbol{Q}_k) \\ \boldsymbol{z}_k &= \boldsymbol{H}_k \boldsymbol{x}_k + \boldsymbol{v}_k, \quad \boldsymbol{v}_k \sim \mathcal{N}(0, \boldsymbol{R}_k) \end{aligned}$$

where $\boldsymbol{F}_k$ and $\boldsymbol{H}_k$ represent, respectively, the state update and measurement matrices and $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$ are the process and measurement covariance matrices, respectively. The Kalman recursion [25] is given below:

$$\begin{aligned} \tilde{\boldsymbol{x}}_{k+1|k} &= \boldsymbol{F}_k \hat{\boldsymbol{x}}_{k|k}, \tilde{\boldsymbol{P}}_{k+1|k} = \boldsymbol{F}_k \hat{\boldsymbol{P}}_{k|k} \boldsymbol{F}_k^T + \boldsymbol{Q}_k, \\ \boldsymbol{S}_k &= \boldsymbol{R}_k + \boldsymbol{H}_k \tilde{\boldsymbol{P}}_{k+1|k} \boldsymbol{H}_k^T, \tilde{\boldsymbol{K}}_k = \tilde{\boldsymbol{P}}_{k+1|k} \boldsymbol{H}_k^T \boldsymbol{S}_k^{-1}, \\ \hat{\boldsymbol{x}}_{k+1|k+1} &= \tilde{\boldsymbol{x}}_{k+1|k} + \tilde{\boldsymbol{K}}_k (\boldsymbol{z}_k - \boldsymbol{H}_k \tilde{\boldsymbol{x}}_{k+1|k}), \\ \hat{\boldsymbol{P}}_{k+1|k+1} &= (\boldsymbol{I} - \tilde{\boldsymbol{K}}_k \boldsymbol{H}_k) \tilde{\boldsymbol{P}}_{k+1|k}, \end{aligned}$$

where $\tilde{\cdot}$ and $\hat{\cdot}$ denote the predicted and the estimated vector / matrix, respectively and $\tilde{\boldsymbol{K}}$ represents the Kalman gain.

Acknowledgment

We acknowledge the support from the Pakistan Air Force (Govt. of Pakistan) and the Dept. of ACSE (University of Sheffield) for granting scholarships to the first author.

References

[1] S. Blackman and R. Popoli, "Design and analysis of modern tracking systems (book)," *Norwood, MA: Artech House.*, 1999.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[3] 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, July 2008.

[4] M. Feldmann, D. Franken, 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, April 2011.

[5] 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 Processing*, vol. 25, pp. 1–16, 2014.

[6] K. Granstrom, M. Baum, and S. Reuter, "Extended object tracking: Introduction, overview and applications," *ISIF Journal of Advances in Information Fusion*, vol. 12, no. 2, pp. 139–174, 2017.

[7] Y. Xie, "A range-dependent echo-association algorithm and its application in split-beam sonar tracking of migratory salmon in the fraser river watershed," *IEEE Journal of Oceanic Engineering*, vol. 25, no. 3, pp. 387–398, 2000.

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

[9] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, X. Zhao, and T.-K. Kim, "Multiple object tracking: A literature review," *arXiv preprint arXiv:1409.7618*, 2014.

[10] C. E. Rasmussen and C. K. Williams, *Gaussian processes for machine learning*. MIT press Cambridge, 2006, vol. 1.

[11] J. Hartikainen and S. Särkkä, "Kalman filtering and smoothing solutions to temporal Gaussian process regression models," in *IEEE International Workshop on Machine Learning for Signal Processing*. IEEE, 2010, pp. 379–384.

[12] T. D. Barfoot, C. H. Tong, and S. Särkkä, "Batch continuous-time trajectory estimation as exactly sparse Gaussian process regression." in *Robotics: Science and Systems*. Citeseer, 2014.

[13] J. Dong, B. Boots, and F. Dellaert, "Sparse Gaussian processes for continuous-time trajectory estimation on matrix lie groups," *arXiv preprint arXiv:1705.06020*, 2017.

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

[15] W. Aftab, A. De Freitas, M. Arvaneh, and L. Mihaylova, "A Gaussian process approach for extended object tracking with random shapes and for dealing with intractable likelihoods," in *Proceedings of the 22nd Conference on Digital Signal Processing (DSP)*. IEEE, 2017, pp. 1–5.

[16] ——, "A Gaussian process convolution particle filter for multiple extended objects tracking with non-regular shapes," in *Proceedings of the 21st International Conference on Information Fusion (FUSION)*. IEEE, 2018, pp. 1–8.

[17] W. Aftab, R. Hostettler, A. De Freitas, M. Arvaneh, and L. Mihaylova, "Spatio-temporal Gaussian process models for extended and group object tracking with irregular shapes," *IEEE Transactions on Vehicular Technology*, 2019.

[18] M. Lázaro-Gredilla, S. Van Vaerenbergh, and N. D. Lawrence, "Overlapping mixtures of Gaussian processes for the data association problem," *Pattern Recognition*, vol. 45, no. 4, pp. 1386–1395, 2012.

[19] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking: decision-based methods," in *Signal and Data Processing of Small Targets 2002*, vol. 4728. International Society for Optics and Photonics, 2002, pp. 511–535.

[20] ——, "Survey of maneuvering target tracking. Part V. Multiple-model methods," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1255–1321, 2005.

[21] Y. Bar-Shalom and X.-R. Li, "Estimation and tracking- Principles, techniques, and software," *Norwood, MA: Artech House, Inc.*, 1993.

[22] R. A. Singer, "Estimating optimal tracking filter performance for manned maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, no. 4, pp. 473–483, 1970.

[23] K. Mehrotra and P. R. Mahapatra, "A jerk model for tracking highly maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 4, pp. 1094–1105, 1997.

[24] J. P. Helferty, "Improved tracking of maneuvering targets: The use of turn-rate distributions for acceleration modeling," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 32, no. 4, pp. 1355–1361, 1996.

[25] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[26] D. Magill, "Optimal adaptive estimation of sampled stochastic processes," *IEEE Transactions on Automatic Control*, vol. 10, no. 4, pp. 434–439, 1965.

[27] D. Lainiotis, "Optimal adaptive estimation: Structure and parameter adaption," *IEEE Transactions on Automatic Control*, vol. 16, no. 2, pp. 160–170, 1971.

[28] H. A. Blom, *A sophisticated tracking algorithm for ATC surveillance radar data*. NLR, 1985.

[29] X. R. Li and Y. Bar-Shakm, "Mode-set adaptation in multiple-model estimators for hybrid systems," in *Proceedings of the American Control Conference*. IEEE, 1992, pp. 1794–1799.

[30] P. Boyle, "Gaussian processes for regression and optimisation," 2007.

[31] P. S. Swain, K. Stevenson, A. Leary, L. F. Montano-Gutierrez, I. B. Clark, J. Vogel, and T. Pilizota, "Inferring time derivatives including cell growth rates using Gaussian processes," *Nature Communications*, vol. 7, p. 13766, 2016.

[32] O. Rudovic, M. Pantic, and I. Patras, "Coupled Gaussian processes for pose-invariant facial expression recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1357–1369, 2013.