sensors

MDPI

Article

Performance Evaluation of a Maneuver Classification Algorithm Using Different Motion Models in a Multi-Model Framework

Máté Kolat, Olivér Törő and Tamás Bécsi *

Department of Control for Transportation and Vehicle Systems, Budapest University of Technology and Economics, H-1111 Budapest, Hungary; mate.kolat@edu.bme.hu (M.K.); toro.oliver@kjk.bme.hu (O.T.)
* Correspondence: becsi.tamas@kjk.bme.hu

Abstract: Environment perception is one of the major challenges in the vehicle industry nowadays, as acknowledging the intentions of the surrounding traffic participants can profoundly decrease the occurrence of accidents. Consequently, this paper focuses on comparing different motion models, acknowledging their role in the performance of maneuver classification. In particular, this paper proposes utilizing the Interacting Multiple Model framework complemented with constrained Kalman filtering in this domain that enables the comparisons of the different motions models' accuracy. The performance of the proposed method with different motion models is thoroughly evaluated in a simulation environment, including an observer and observed vehicle.

Keywords: filtering; constraints; IMM; maneuver targeting; maneuver classification; motion models

check for updates

Citation: Kolat, M.; Törő, O.; Bécsi, T. Performance Evaluation of a Maneuver Classification Algorithm Using Different Motion Models in a Multi-Model Framework. *Sensors* **2022**, *22*, 347. https://doi.org/10.3390/s22010347

Academic Editors: Giovanni Pau and Felipe Jiménez

Received: 4 November 2021
Accepted: 1 January 2022
Published: 4 January 2022

1. Introduction

Environment perception is vital for safe functionality as the decision-making layer relies solely on its awareness of the given traffic scenario. Based on the current global trend, the number of autonomous vehicles is expected to increase, requiring solutions for new types of problems [1,2]. At present, the level of a considerable number of available sensors, and thus the possible combination of sensor-fusions, can offer a wide scale in cost and robustness [3]. Furthermore, the spread of the different type of driver assistance systems create a demand for cost-effective testing methods in simulated environments [4]. Such methods have significant advantages like cost-effectiveness, test reproducibility, investigating unsafe situations, different weather, and daytime conditions [5]. Vehicle tracking evolved into a critical challenge in the automotive industry since it is necessary to design advanced driver assistance systems such as Adaptive Cruise Control, Emergency Braking System, Blind Spot Detection, or Collision Avoidance.

In the case of vehicle tracking, the next maneuver of the tracked object is unknown; therefore, it cannot be decided in advance which motion model should be used during the estimation, despite that the performance of the estimators relies highly on it [6]. Motion models can be classified by complexity, starting from linear models [7], which are the constant velocity (CV) and constant acceleration (CA) models. These models have limited capabilities for describing more complex motions; however, the probability density function is easier to handle. In the case of linear systems, from the data fusion point of view, Linear Kalman Filter or Kalman Filter (KF) is the most commonly used approach [8]. Constant Turn and Rate Velocity (CTRV) and Constant Turn Rate and Acceleration (CTRA) are more complex curvilinear models considering rotation, where Extended Kalman Filter (EKF) is a suitable solution that can handle nonlinearity. Generally, an appropriate choice of the motion model can significantly increase the vehicle tracking system's performance. However, complexity does not always lead to a better performance [7]. There are many ways to deal with the uncertainty of the used motion model, including the Multiple Model (MM) filtering [9], where numerous filters can be designed and select the best prediction. In other words, it examines all the possible combinations of the predefined models at each timestep and returns with the best estimation. Interacting Multiple Model (IMM), which is a subtype of Multiple Model, is presented by Blom and Bar-Shalom [10] allows different filters to run parallel and described in Section 2.3. Wenkang et al. compared the traditional Square Root Cubature Kalman Filter (SCKF) to their IMM-SCKF algorithm regarding vehicle state estimation [11]. The IMM is originally proposed with Kalman filters [12], however particle filter realization also appeared in the literature [13–18].

The performance of the Kalman filter can be further increased by using state constraints. For example, suppose any boundaries limit the desired safe states. In that case, those limits can be inserted in the filtering process as state constraints and improve its performance with this additional information [19].

Kalman filter can handle state-space representations where both the measurement and the transition equations are linear with Gaussian noise. Vehicle tracking tasks can also be approached by Particle Filters (PF) [20], which is used for nonlinear and non-Gaussian problems [21]. However, it is a computationally much more expensive methodology than the Extended Kalman Filter, and any further increase in the number of particles significantly slows down the data process. Object tracking has extended literature featuring other techniques. Kowlaczuk et al. (2010) used Multiple Model Constrained Filtering with Kalman Filter for air traffic control [22]. Farmer et al. (2002) applied Interacting Multiple Model Kalman Filter for high-speed human motion tracking [23]. Zheng et al. (2009) presented a face detection method using Particle Filter [24], and Kim et al. (2020) introduced a vehicle position tracking method utilizing Lidar and radar measurements with the help of EKF [25]. Zhao et al. (2018) implemented a multi-object tracking algorithm using correlation filter [26]. Rakos et al. (2020) examined and compared various classification methods for lane change tracking [27]. Toro et al. designed a multiple object tracking algorithm, applying a particle filter-based Probability Hypothesis Density filter to track hidden but likely present objects due to occlusion [28].

Even though a significant number of Kalman Filter-based methods can be found that use the pre-fit residual, only a few apply the post-fit residual. Ormsby et al. presented a solution, where they combined the traditional pre-fit and the post-fit residual, which serves as a solution for other Multiple Model methods, like Multiple Model Adaptive Estimator (MMAE) [29]. Henderson et al. considered the carrier-phase integer ambiguity problem in the context of GPS positioning [30]. The authors used a multiple-model approach with different ambiguity hypotheses. For model likelihood computation, the post-fit residual and its covariance were used. Zhang et al. used the post-fit residual to estimate the unknown measurement noise covariance matrix [31]. The utilization of the post-fit residual increases the estimation performance and provides a better prediction for maneuver classification. Moreover, selecting the best motion model is essential, likewise considering a particular maneuver.

Contributions of the Paper

This work examines and compares four motion models (CV, CA, CTRV, and CTRA), considering a maneuver classification method using constrained Kalman filtering with multi-model estimation. The measurements come from radar and camera sources, and the maneuvers of a road vehicle are detected and classified using various complexity of motion models. Later, the models are evaluated and compared based on their performance. The constrained filters are arranged in the structure of the interacting multiple model estimator. Constraints customize each filter to match a specific type of maneuver. The quality of a filter is determined by examining the post-fit residual.

The current study only uses linear constraints, which enables the utilization of the Kalman method. Therefore, the particle filtering approach is not necessary. On the other hand, the primary purpose, to compare the motion models, gives the same problem for all filtering paradigms.

The paper is structured as follows. Section 2 summarizes the theoretical background of the presented method. The simulation framework for testing and the maneuver detection methods are presented in Section 3. Section 4 examines the performance of the estimator. Concluding remarks are given in Section 5.

2. Methodology

In this study, multiple motion models are used for performance comparison purposes. Therefore, the considered system is described in double measure concerning the motion model; thus, the system can be described by a linear discrete-time dynamic model or a nonlinear model [19]. The described model using linear state transition with time index k:

$$\mathbf{x}_k = F_k \mathbf{x}_{k-1} + \mathbf{w}_k \tag{1}$$

$$\mathbf{z}_k = H_k \mathbf{x}_k + \mathbf{v}_k \tag{2}$$

where $\mathbf{x}_k \in \mathbb{R}^{n_x}$ is the state vector, derived from the state transitions matrix $F_k \in \mathbb{R}^{n_x \times n_x}$ and $\mathbf{w}_k \in \mathbb{R}^{n_w}$ process noise. $\mathbf{z}_k \in \mathbb{R}^{n_z}$ is achieved through the observation model $Hx_k \in \mathbb{R}^{n_z \times n_x}$ and observation noise $\mathbf{v}_k \in \mathbb{R}^{n_z}$. Both $\mathbf{w}_k$ and $\mathbf{v}_k$ is assumed to be a zero-mean Gaussian noise, with covariance Q_k: $\mathbf{w}_k \sim \mathcal{N}(0, Q_k)$ and R_k: $\mathbf{v}_k \sim \mathcal{N}(0, R_k)$. The described model uses nonlinear state transition with time index k:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{w}_k \tag{3}$$
$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{v}_k \tag{4}$$

where the new predicted state $\mathbf{x}_{k+1}$ is generated from function f and the measurement prediction $\mathbf{z}_k$ is calculated by function h.

2.1. Kalman Filter

Kalman Filter, which is the base of this study, is a linear recursive estimator [32]. For nonlinear filtering problems, the Extended Kalman Filter is an effective solution, applying the linearization through the computation of the Jacobian as the partial derivatives of the matrices. The partial derivatives of F_k and H_k:

$$F_k = \left.\frac{\partial f}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{k-1|k-1}} \tag{5}$$

$$H_k = \left.\frac{\partial h}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{k-1|k-1}} \tag{6}$$

The essential concept of the Kalman Filter is the following:

$$\hat{\mathbf{x}}_{k|k-1} = F_k \hat{\mathbf{x}}_{k-1|k-1} \tag{7}$$
$$P_{k|k-1} = F_k P_{k-1|k-1} F^T + G_k Q_k G_k^T \tag{8}$$
$$\mathbf{r}_{k|k-1} = \mathbf{z}_k - H_k \hat{\mathbf{x}}_{k|k-1} \tag{9}$$
$$S_{k|k-1} = H_k P_{k|k-1} H_k^T + R_k \tag{10}$$
$$K_k = P_{k|k-1} H_k^T S_{k|k-1}^{-1} \tag{11}$$
$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k \mathbf{r}_{k|k-1} \tag{12}$$
$$P_{k|k} = (\mathbf{I} - K_k H_k) P_{k|k-1} \tag{13}$$

In (7) and (8), the a priori state estimate and the corresponding state covariance is computed, while (9) and (10) define the pre-fit residual and its covariance, where R_k denotes the measurement noise. In (11), the Kalman gain is calculated. The *a posteriori* state estimate and error covariance are given by (12) and (13) For the Extended Kalman Filter (7) and (9) should be replaced with

$$\hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}) \tag{14}$$

$$\mathbf{r}_{k|k-1} = \mathbf{z}_k - h(\hat{\mathbf{x}}_{k|k-1}) \tag{15}$$

The estimation quality is defined by a zero-mean PDF, using the pre-fit residual and its covariance:

$$\Lambda = \mathcal{N}(\mathbf{r}, 0, S) \tag{16}$$

The estimation quality can also be defined with the help of the post-fit residual and its covariance:

$$\mathbf{r}_{k|k} = \mathbf{z}_k - H_k\hat{\mathbf{x}}_{k|k} \tag{17}$$

$$S_{k|k} = (\mathbf{I} - H_k K_k)S_{k|k-1}(\mathbf{I} - H_k K_k)^T \tag{18}$$

2.2. Constrained Filtering

The Kalman filtering method is one of the most acknowledged filtering approaches; however, occasionally, it is not robust enough [33]. In different cases, like maneuver classification, filter categorization can be a necessary solution that is reachable using constraints, where the system is conditional to it, of which estimation can violate those, in case of the inadequacy of integration of constraints in the system model or the filtering process. There are numerous approaches to integrating the system model's constraints or filtering methods, such as measurement augmentation or estimation projection. Constrained filtering can utilize this additional information and produce classes of outputs. The classification can be achieved using defined upper or lower bounds. Constraints are differentiated in several ways: equality or non-equality constraints, linear or nonlinear, soft or hard.

Linear equality constraints are formulated as follows:

$$D_k\mathbf{x}_k = \mathbf{d}_k \tag{19}$$

and non-equality constraints:

$$D_k\mathbf{x}_k \leq \mathbf{d}_k \tag{20}$$

where $\mathbf{d}_k \in \mathbb{R}^{n_c}$ is the constraint vector and $D_k \in \mathbb{R}^{n_c \times n_x}$ is the constraint matrix.

2.2.1. Estimation Projection

One of the solutions to the problem is the estimation projection [19]. It investigates even the state estimate fulfill a defined constraint. In this case, the state is projected in the constrained space, which is formulated as follows:

$$\hat{\mathbf{x}}^d_{k|k} = \arg\min_{\mathbf{x}} \left(\mathbf{x} - \hat{\mathbf{x}}_{k|k}\right)^\top W \left(\mathbf{x} - \hat{\mathbf{x}}_{k|k}\right) \tag{21}$$

where $\hat{\mathbf{x}}^d_{k|k}$ is the constrained state estimate, $\hat{\mathbf{x}}_{k|k}$ is the unconstrained updated estimate, and W is a positive definite symmetric weighting matrix, which is chosen as $P^{-1}_{k|k}$; thus, the form of the estimation is

$$\hat{\mathbf{x}}^d_{k|k} = \hat{\mathbf{x}}_{k|k} + K^p_k(\mathbf{d}_k - D_k\hat{\mathbf{x}}_{k|k}) \tag{22}$$

where K^p_k is

$$K^p_k = P_{k|k}D^T_k(D_k P_{k|k} D^T_k)^{-1} \tag{23}$$

and the covariance of the constrained state estimate is

$$P^d_{k|k} = P_{k|k} - K^d_k D_k P_{k|k} \tag{24}$$

Estimation projection can handle non-equality constraints filtering problems using the form of (20). The rows of D and d corresponding to the active constraints are chosen if x violates those; thus, a newly formulated constraint matrix and vector are created, with which (21) can be solved:

$$D_k \hat{\mathbf{x}}_{k|k} = \mathbf{d}_k \tag{25}$$

2.2.2. Measurement Augmentation

The other used solution for equality constraint state estimation is measurement augmentation [34]. Equality constraint (19) can be extended by additional noise (δ_k), considering the constraints as soft constraints; thus, it has to be satisfied relatively:

$$D_k \mathbf{x}_k = \mathbf{d}_k + \delta_k \tag{26}$$

The augmented measurement equation is

$$\begin{bmatrix} \mathbf{z}_k \\ \mathbf{d}_k \end{bmatrix} = \underbrace{\begin{bmatrix} H_k \\ D_k \end{bmatrix}}_{H_k^d} \mathbf{x}_k + \begin{bmatrix} \mathbf{v}_k \\ \delta_k \end{bmatrix} \tag{27}$$

or in shorter form:

$$\mathbf{z}_k^d = H_k^d \mathbf{x}_k + \mathbf{v}_k^d \, . \tag{28}$$

The covariance of the augmented noise term $\mathbf{v}_k^d$ is R_k^d. When the noise term δ_k is zero, it is called Perfect Measurements (PM) and handled as hard constraints. The equations are the same as the Kalman Filter equations, but with augmented elements:

$$\mathbf{r}_{k|k-1}^d = \mathbf{z}_k^d - H_k^d \hat{\mathbf{x}}_{k|k-1} \tag{29}$$

$$S_k^d = H_k^d P_{k|k-1} (H_k^d)^T + R_k^d \tag{30}$$

$$K_k^d = P_{k|k-1} (H_k^d)^T + R_k^d \tag{31}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k^d r_{k|k-1}^d \tag{32}$$

$$P_{k|k} = (\mathbf{I} - K_k^d H_k^d) P_{k|k-1} \tag{33}$$

2.3. Multi-Model Estimation

The multiple model (MM) approach helps to reduce the uncertainties of the model, which assumes that the system behaves according to one of a finite number of models [9]. The fundamental concept of the multiple model approach is to run the designed models parallel and select one with the highest performance [35]. In this study, the significant uncertainties are regarding motion models and sensors considering maneuver tracking problems.

Each model calculates its likelihood based on its constrained post-fit residual and the corresponding covariance designed using (22)–(24) in case of estimation projection as follows:

$$\mathbf{r}_{k|k}^d = \mathbf{z}_k - H_k \hat{\mathbf{x}}_{k|k}^d \tag{34}$$

where $\mathbf{r}_{k|k}^d$ is the constrained post-fit residual. The corresponding covariance is formulated as follows:

$$S_{k|k}^d = S_{k|k} + H_k K_k^d D_k P_{k|k}^{-1} H_k^T \tag{35}$$

The zero-mean PDF is defined with the help of (16). In the case of measurement augmented constraints, (29) and (30) are used for probability estimation.

2.4. Motion Models

Vehicle tracking must provide a decent position estimation. The two assets of the motion models predict the vehicle's future position and describe the dynamic behavior [7].

The performance of the estimator relies on the type of the motion model; thus, it is also feasible to compare motion models for certain applications [6]. However, it is challenging to select an appropriate motion model in advance. Motion models can be classified based on their complexity. In this study, four distinctive motion models are applied: *Constant Velocity* (CV) and *Constant Acceleration* (CA) models, which are linear models, and *Constant Turn Rate and Velocity* (CTRV) and *Constant Turn Rate and Acceleration* (CTRA), which are curvilinear models (see Figure 1).

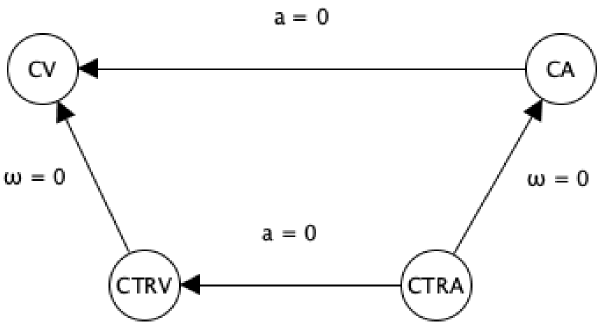

Figure 1. The applied motion models and their relations.

The lowest level motion models based on their complexities are *Constant Velocity* (CV) and *Constant Acceleration* (CA) models. Linear models have the benefit of ensuring a great state probability distribution, yet both of these models presume straight motions only; thus, the state vectors are as follows:

$$\mathbf{x}_{cv} = (x, y, \theta, v, \omega)^T \tag{36}$$

$$\mathbf{x}_{ca} = (x, y, \theta, a, v, \omega)^T \tag{37}$$

where the acceleration a can be derived from the velocity v, of which lateral and longitudinal components are calculated using the heading angle θ. Yaw rate ω is ignored in linear models. Curvilinear motion models *Constant Turn Rate and Velocity* (CTRV) and *Constant Turn Rate and Acceleration* (CTRA) are on a level above, which can take into account the rotations. The state vectors are as follows:

$$\mathbf{x}_{ctrv} = (x, y, \theta, v, \omega)^T \tag{38}$$

$$\mathbf{x}_{ctra} = (x, y, \theta, a, v, \omega)^T \tag{39}$$

However, for the utilization of these models, due to their nonlinearity, a filter is needed, which can manage it. Therefore, in this study, the previously mentioned *Extended Kalman Filter* (EKF) is used for that objective.

3. Evaluation

This section presents the case study of the maneuver classification method using constrained Kalman filtering and IMM. Figure 2 shows the flowchart of the constrained IMM filter. The built scenario, the actors, the established trajectory, and the simulation environment is described in Section 3.1. The constraints corresponding to the maneuvers are introduced in Section 3.2 and the type of the constraints likewise. The estimation method and the used measurements are conferred in this subsection additionally.

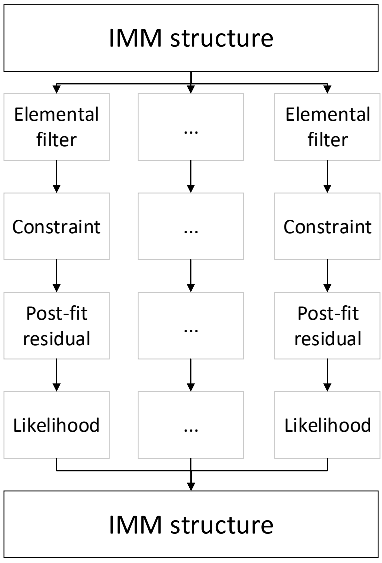

Figure 2. Dataflow of the proposed method.

Figure 3. Observer (**rear**) and maneuvering vehicle (**front**) in the simulation.

3.1. Environment

The study is implemented in Simulink, using Automated Driving Toolbox provided by Matlab. In the scenario, two actors take place (see Figure 3). The observed vehicle moves along a predefined trajectory performing various maneuvers. The observer moves behind and predicts the maneuvers using radar and camera measurements. Two types of maneuvers can be distinguished corresponding to the lateral motion and the longitudinal, are described in Tables 1 and 2.

Table 1. Lateral maneuvers of the observed vehicle.

Maneuver	Start Time [s]	End Time [s]
Left lane	0	5
Lane change	5.1	6.4
Right lane	6.5	11.7
Lane change	11.8	12.6
Left lane	12.7	18.4
Lane change	18.5	19
Right lane	19.1	23.2
Lane change	23.3	23.9
Left lane	24	26.6

Table 2. Longitudinal maneuvers of the observed vehicle.

Speed	Start Time [s]	End Time [s]
Gaining distance	0	1.5
Collision warning	1.6	3.1
Distance keeping	3.2	4.1
Losing distance	4.2	8.5
Distance keeping	8.6	24
Losing distance	24	26.6

The observer vehicle moves behind and takes measures via radar. Moreover, the observer vehicle also detects the lane line using camera information; therefore, it can describe the used lane line. The radar takes measurements in polar coordinates; thus, the collected information is the angle γ, velocity v, and distance r between the two actors. The state space is represented in various forms based on the used motion model. Four types of motion models are used and compared in this study: the previously mentioned CV, CA, CTRV, and CTRA. The state-space representations are described in (36)–(39); therefore, the radar measurements $\mathbf{z}_r = [\gamma, v, r]$ transformation is mandatory to be consistent with the state vector during the filtering process. The distance part of the measurement vector and the state-vector is not identical; therefore, the associated part of the covariance matrix $R_1 = \text{diag}(\sigma_\theta^2, \sigma_r^2)$ is needed to be transformed likewise. The corresponding covariance is calculated using the Jacobian polar to Cartesian transformation:

$$R^p = JR_1J^T \tag{40}$$

$$J = \begin{bmatrix} -r\sin\theta & \cos\theta \\ r\cos\theta & \sin\theta \end{bmatrix} \tag{41}$$

and

$$R^p = \begin{bmatrix} \sigma_r^2\cos^2\theta + \sigma_\theta^2 r^2\sin\theta & (\sigma_r^2 - \sigma_\theta^2 r^2)\cos\theta\sin\theta \\ (\sigma_r^2 - \sigma_\theta^2 r^2)\cos\theta\sin\theta & \sigma_\theta^2 r^2\cos^2\theta + \sigma_r^2\sin\theta \end{bmatrix} \tag{42}$$

The yaw rate is derived from camera information, which allows detecting lane line as a polynomial. Camera measures the curvature derivative $\dot{\kappa}$, curvature κ, the heading angle θ and the lateral offset y_{lat}; thus, $z^c = [\dot{\kappa}, \kappa, \theta, y_{lat}]$. Yaw rate ω is calculated using the camera measurement of curvature derivative and curvature. Moreover, the distance measure of the radar sensor is mandatory likewise.

Yaw rate ω is calculated as follows:

$$\omega = \dot{\rho}v \tag{43}$$

where v is the doppler velocity and $\dot{\rho}$ is

$$\dot{\rho} = \dot{\kappa}x + \kappa \tag{44}$$

where x is the longitudinal component of the distance measurement.

For each motion model, position error and maneuver probabilities are calculated. The maneuver probabilities are collected using IMM, which allows running the corresponding filters parallel, where each filter exemplifies a constrained specified maneuver. The filters are implemented in Simulink. The state transition matrices F and matrices H are the following, where $T = 0.1$ s is the sampling time:

$$F^{CV} = \begin{bmatrix} 1 & 0 & -Tv\sin\theta & T\cos\theta & 0 \\ 0 & 1 & Tv\cos\theta & T\sin\theta & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{45}$$

$$H^{CV} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{46}$$

$$F^{CA} = \begin{bmatrix} 1 & 0 & -Tv\sin\theta & \frac{T^2}{2}\cos\theta & T\cos\theta & 0 \\ 0 & 1 & Tv\cos\theta & \frac{T^2}{2}\sin\theta & T\sin\theta & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & T & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{47}$$

$$H^{CA} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{48}$$

The yaw rate ω is ignored by the constant velocity and constant acceleration motion models; hence, the 0 value does not evolve through the state transition matrix F and matrix H. Furthermore, heading angle θ and acceleration a are not measured, only predicted; thus, ignored likewise in matrix H.

The state transition F and measurement H matrices for the CTRV model are

$$F^{CTRV} = \begin{bmatrix} 1 & 0 & \frac{\partial f_x^{CTRV}}{\partial\theta} & \frac{\partial f_x^{CTRV}}{\partial v} & \frac{\partial f_x^{CTRV}}{\partial\omega} \\ 0 & 1 & \frac{\partial f_y^{CTRV}}{\partial\theta} & \frac{\partial f_y^{CTRV}}{\partial v} & \frac{\partial f_y^{CTRV}}{\partial\omega} \\ 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{49}$$

where the partial derivatives are

$$\frac{\partial f_x^{CTRV}}{\partial\theta} = \frac{v}{\omega}\cos(\theta + \omega T) - \frac{v}{\omega}\cos\theta \tag{50}$$

$$\frac{\partial f_x^{CTRV}}{\partial v} = \frac{\sin(\theta + \omega T) - \sin\theta)}{\omega} \tag{51}$$

$$\frac{\partial f_x^{CTRV}}{\partial\omega} = \frac{v}{\omega^2}(\cos(\theta + \omega T) + \omega T\sin(\theta + \omega T) - \cos\theta) \tag{52}$$

$$\frac{\partial f_y^{CTRV}}{\partial\theta} = \frac{v}{\omega}\sin(\theta + \omega T) - \frac{v}{\omega}\sin\theta \tag{53}$$

$$\frac{\partial f_y^{CTRV}}{\partial v} = \frac{\cos\theta - \cos(\theta + \omega T)}{\omega} \tag{54}$$

$$\frac{\partial f_y^{CTRV}}{\partial\omega} = \frac{v}{\omega^2}(\cos(\theta + \omega T) + \omega T\sin(\theta + \omega T) - \cos\theta) \tag{55}$$

and

$$H^{CTRV} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{56}$$

The state transition F and measurement H matrices for the CTRA model are

$$F^{CTRA} = \begin{bmatrix} 1 & 0 & \frac{\partial f_x^{CTRA}}{\partial\theta} & \frac{\partial f_x^{CTRA}}{\partial a} & \frac{\partial f_x^{CTRA}}{\partial v} & \frac{\partial f_x^{CTRA}}{\partial\omega} \\ 0 & 1 & \frac{\partial f_y^{CTRA}}{\partial\theta} & \frac{\partial f_y^{CTRA}}{\partial a} & \frac{\partial f_y^{CTRA}}{\partial v} & \frac{\partial f_y^{CTRA}}{\partial\omega} \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & T & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{57}$$

where the partial derivatives are

$$\frac{\partial f_x^{CTRA}}{\partial\theta} = \frac{a(\sin\theta - \sin(\theta+\omega T)) + \omega(aT+v)\cos(\theta+\omega T) - v\omega\cos\theta}{\omega^2} \tag{58}$$

$$\frac{\partial f_x^{CTRA}}{\partial a} = \frac{-\cos\theta + T\omega\sin(\theta+\omega T) + \cos(\theta+\omega T)}{\omega^2} \tag{59}$$

$$\frac{\partial f_x^{CTRA}}{\partial v} = \frac{\sin(\theta+\omega T) - \sin\theta}{\omega} \tag{60}$$

$$\frac{\partial f_x^{CTRA}}{\partial\omega} = \frac{2a\cos\theta + (T\omega^2(aT+v) - 2a)\cos(\theta+\omega T) - \omega(2aT+v))\sin(\theta+\omega T) + v\omega\sin\theta}{\omega^3} \tag{61}$$

$$\frac{\partial f_y^{CTRA}}{\partial\theta} = \frac{-a(\cos\theta + a\cos(\theta+\omega T)) + \omega(aT+v)\sin(\theta+\omega T) + v\sin(\theta+\omega T) - v\sin\theta}{\omega^2} \tag{62}$$

$$\frac{\partial f_y^{CTRA}}{\partial a} = -\frac{\sin\theta + T\omega\cos(\theta+\omega T) - \sin(\theta+\omega T)}{\omega^2} \tag{63}$$

$$\frac{\partial f_y^{CTRA}}{\partial v} = \frac{\cos\theta - \cos(\theta+\omega T)}{\omega} \tag{64}$$

$$\frac{\partial f_y^{CTRA}}{\partial\omega} = \frac{2a\sin\theta + (T\omega^2(aT+v) - 2a)\sin(\theta+\omega T) + \omega(2aT+v))\cos(\theta+\omega T) - v\omega\cos\theta}{\omega^3} \tag{65}$$

and

$$H^{CTRA} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{66}$$

Curvilinear motion models (CTRV, CTRA) allow yaw rate and require nonlinear filters; therefore, Extended Kalman Filter is used. Thus, linearization is mandatory, which is described above. The heading angle θ and acceleration a is ignored likewise in the case of linear models. The noise for the motion model is a discrete white noise, which is denoted by covariance Q:

$$Q = \Gamma q \Gamma^T \tag{67}$$

where Γ is the known disturbance matrix:

$$\Gamma = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & T & \frac{T^2}{2} \\ T & 0 & 0 & 0 \\ \frac{T^2}{2} & T & 0 & 0 \\ 0 & 0 & 0 & T \end{bmatrix} \tag{68}$$

and T is the sampling interval.

3.2. Constraints

Three lateral (Table 3) and four longitudinal (Table 4) maneuvers are introduced in this study. In the constraints of the lateral maneuvers (Table 5), upper and lower bounds are applied to the lateral motions as follows:

Table 3. Constraints for lateral maneuvers.

Mode	Position Constraint	Velocity Constraint
Right lane	$y < l - 0.5$	$\dot{y} \,\epsilon\, \mathcal{N}(0,1)$
Left lane	$y > l + 0.5$	$\dot{y} \,\epsilon\, \mathcal{N}(0,1)$
Lane change	$l - 0.5 < y < l + 0.5$	$\dot{y} \,\epsilon\, \mathcal{N}(0,1)$

where l denotes the lane width. These maneuvers define even the observed vehicle moves in lane or performing lane change. The constraint corresponding to the right lane consists of an upper limit, calculated by the width of the lane line. The left lane has lower limits, and the lane change maneuver constraint consists of upper and lower limits. The position constraints are introduced using hard inequality constraints and estimate projection. In contrast, the velocity component is derived as a zero-mean Gaussian because applying limitation is not required.

Four distinctive longitudinal maneuvers are introduced in the study based on the distance and velocity difference as follows:

Table 4. Longitudinal maneuvers.

Maneuver	Velocity Constraint	Distance Constraint
Losing distance	$\dot{x} <= -1$	$x > 10$
Gaining distance	$\dot{x} >= 1$	$x \,\epsilon\, \mathbb{R}^+$
Distance keeping	$\dot{x} = 0$	$x \,\epsilon\, \mathbb{R}^+$
Collision warning	$\dot{x} <= -1$	$x <= 10$

Table 5. Constraint type of lateral maneuvers.

Maneuver	Type of Constraint	Estimation Method
Right lane	Hard inequality Soft equality	Estimate projection Measurement augmentation
Left lane	Hard inequality Soft equality	Estimate projection Measurement augmentation
Lane change	Hard inequality Soft equality	Estimate projection Measurement augmentation

Velocity and distance constraints are applied in various longitudinal maneuvers. Losing distance occurs when the observer is faster with at least 1 $\mathrm{m} \cdot \mathrm{s}^{-1}$ and the distance between the two actors is more than 10 m. Collision warning arises when the velocity difference is minimum -1 $\mathrm{m} \cdot \mathrm{s}^{-1}$, and the distance is maximum 10 m. Distance keeping is implemented as a measurement augmented soft constraint; thus, the estimation is not required to be explicitly zero, only approximately. Gaining distance occurs when the velocity difference is at least 1 $\mathrm{m} \cdot \mathrm{s}^{-1}$.

The longitudinal constraints are implemented as follows (Table 6):

Table 6. Constraint type of longitudinal maneuvers.

Maneuver	Type of Constraint	Estimation Method
Losing distance	Hard inequality Soft equality	Estimate projection Measurement augmentation
Gaining distance	Hard inequality Soft equality	Estimate projection Measurement augmentation
Distance keeping	Soft equality	Measurement augmentation
Collision warning	Hard inequality	Estimate projection

4. Results

The initial velocity of the observed vehicle is 30 $\mathrm{m \cdot s^{-1}}$ and speed up subsequently to 35 $\mathrm{m \cdot s^{-1}}$. The observer vehicle has an initial velocity of 35 $\mathrm{m \cdot s^{-1}}$, which firstly slows down to 30 $\mathrm{m \cdot s^{-1}}$, then accelerates back to 35 $\mathrm{m \cdot s^{-1}}$. The motion models and the corresponding filters are evaluated in two aspects. Generally, seven filters run parallel, separated into two classes. Three filters agree to the lateral maneuvers and four to the longitudinals using the same motion models, and each model is evaluated regarding maneuver classification and compared. Moreover, each motion model is assessed by position and velocity error likewise. The mode transition probabilities are designed as

$$\pi_{lat} = \begin{bmatrix} 0.7 & 0.2 & 0.1 \\ 0.15 & 0.15 & 0.7 \\ 0.1 & 0.2 & 0.7 \end{bmatrix} \tag{69}$$

where π_{lat} is the mode transition matrix corresponding to the lateral maneuvers. The first row and column are associated with the progressing in the left lane, the second is the lane changing maneuver, and the last is the right lane. The mode transition matrix regarding the longitudinal maneuvers π_{lon} is designed as follows:

$$\pi_{lon} = \begin{bmatrix} 0.67 & 0.01 & 0.31 & 0.01 \\ 0.01 & 0.58 & 0.11 & 0.30 \\ 0.21 & 0.21 & 0.57 & 0.01 \\ 0.01 & 0.21 & 0.21 & 0.57 \end{bmatrix} \tag{70}$$

Finding the correct parameters for the transition matrix always represents a significant concern. Consequently, trial and error based heuristic searches are conducted to capture the proper parameters for our research.

Losing distance corresponding to the first row and column corresponds to a gaining distance maneuver associated with the second, following the distance keeping prerogative. The last row and column are responsible for collision warnings. The probabilities of the observed maneuvers using the CTRA motion model can be seen in Figure 4. The left side of the figure shows the probabilities regarding the lateral maneuvers. The right side is corresponding the longitudinals.

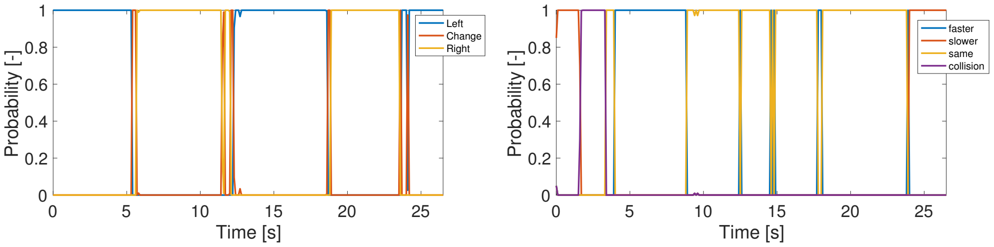

Figure 4. Probabilities of the investigated maneuvers using CTRA model.

The accuracy of the model prediction is computed as follows. The most probable model for each motion model is considered the estimated maneuver at each timestep. These values, which are weighted by the mode probability, are compared to the predefined maneuvers—described in Tables 1 and 2, resulting a prediction accuracy for each motion model. Table 7 describes the model accuracy of the diverse motion models comparing these. The first row denotes the classification precision of the different models regarding the longitudinal maneuver, while the second highlights laterals. The parameters are calculated as the ratio of time when the algorithm correctly classifies the maneuver.

Table 7. Accuracy of the models.

	CV	**CTRV**	**CA**	**CTRA**
lon	93.09%	91.73%	94.06%	93.91%
lat	89.59%	89.58%	89.74%	91.28%

The position error of the unconstrained filters using a specific motion model is calculated in the Euclidean space and presented in Figures 5–8. Comparing the CV and the CTRV motion models, allowing yaw rate assures a better estimation, thus lower error.

The unconstrained estimators' root mean square error (RMSE) is applied on both position components and presented in Table 8.

Table 8. RMSE of the models.

	CV	**CTRV**	**CA**	**CTRA**
X	1.4684	0.6427	0.4297	0.3626
Y	1.8002	0.6139	1.1480	0.8427

The two curvilinear motion models overcome the nonlinear models. CTRA and CTRV models present the best estimation with approximately 0.5 m error in the X and Y direction. In contrast, the estimation of the CV model comprises 1.5 m error. Even though the curvilinear motion models return with the best estimate, thus the lowest error, it is perceivable in Table 8 that the difference is negligible considering maneuver classification. As mentioned above in Section 3.2, constraints help reduce the uncertainty of a system, which is observable in this study likewise. Comparing CA and CTRA motion models, the above statement applies her too, although with lower contrast.

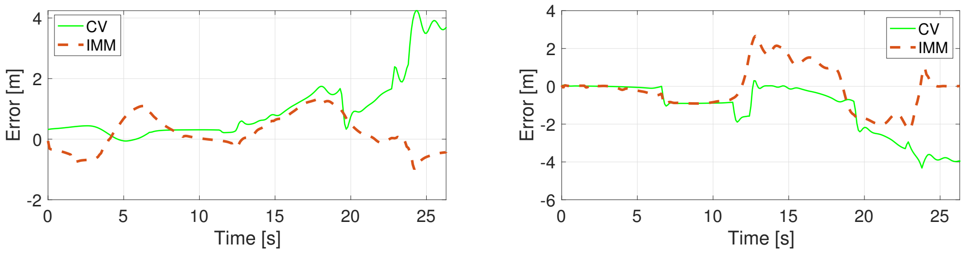

Figure 5. Position error in direction X (**left**) and Y (**right**) using the CV model.

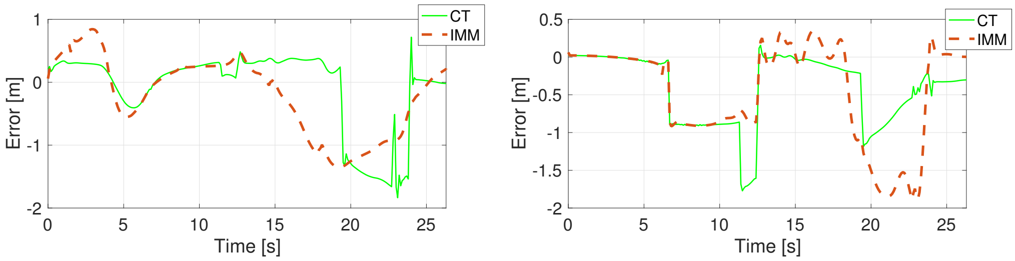

Figure 6. Position error in direction X (**left**) and Y (**right**) using the CTRV model.

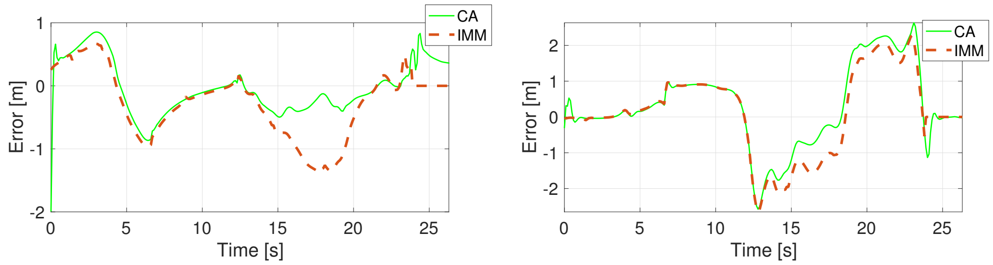

Figure 7. Position error in direction X (**left**) and Y (**right**) using the CA model.

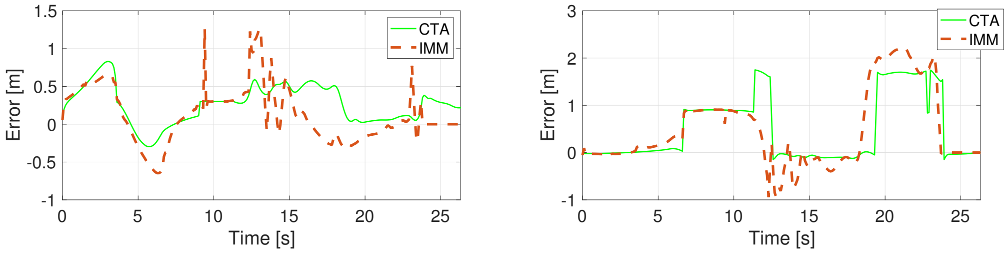

Figure 8. Position error in direction X (**left**) and Y (**right**) using the CTRA model.

5. Conclusions

This paper presents a scenario-independent maneuver classification method, and its main purpose is to evaluate four different motion models in the given framework. The measurement and the state vectors must match, while the constraints must be defined separately. The presented algorithm utilizes the post-fit residual that profoundly enhances the accuracy of maneuver prediction. It is also combined with the IMM framework that enables the evaluation and comparison of the different motion models in a parallel manner. The algorithm uses constrained filters in the IMM structure and detects various maneuvers of an observed vehicle. The results of the maneuver classification are examined and evaluated based on four different motion models. It highlights the idea that from the aspect of the RMSE it is crucial to use the suitable motion model as the curvilinear models provide a significantly better position estimation than the linear. However, it is comprehensible that there is a modest difference between the models when constraints are applied as they alone reduce model uncertainty, as mentioned earlier.

A flexible solution is implemented; thus, the algorithm can be efficiently extended with new maneuvers. The IMM can face problems when the dimension of the various state spaces are not identical. In this study, a dimension extension method is introduced, where the state vector is extended to the same dimension with zero value. An other solution is mixing different dimension state vectors; thus, a more flexible system is implemented [36]. With nonlinear constraints more complex and refined maneuvers could be defined [33]. As the Kalman Filter has limited capabilities to handle nonlinearities, particle filter-based methods could be used instead to perform the filtering task [37]. The Variable Structure IMM (VSIMM) can handle variable model sets adaptively; thus, it can improve the system's performance [38]. Baxter et al. [39] applied an adaptive motion model to track a person based on head-pose. Likewise, switching the motion model adaptively considering the performing maneuver can increase the system's performance.

Author Contributions: Conceptualization, T.B. and M.K.; methodology, O.T.; software, M.K.; validation, O.T.; resources, T.B.; writing—original draft preparation, M.K. and O.T.; writing—review and editing, T.B.; visualization, M.K.; supervision, T.B. All authors have read and agreed to the published version of the manuscript.

Funding: The research is supported by the Ministry of Innovation and Technology NRDI Office within the framework of the Autonomous Systems National Laboratory Program. The research is also supported by the Hungarian Government and co-financed by the European Social Fund through the project "Talent management in autonomous vehicle control technologies" (EFOP-3.6.3-VEKOP-16-2017-00001).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Source and data are available at the authors.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CV	Constant Velocity
CA	Constant Acceleration
KF	Kalman Filter
CTRV	Constant Turn Rate and Velocity
CTRA	Constant Turn Rate and Acceleration
EKF	Extended Kalman Filter
MM	Multiple Model
IMM	Interacting Multiple Model
PF	Particle Filter
PDF	Probability Density Function
PM	Perfect Measurement
RMSE	Root Mean Square Error
VSIMM	Variable Structure Interacting Multiple Model

References

1. Tettamanti, T.; Varga, I.; Szalay, Z. Impacts of autonomous cars from a traffic engineering perspective. *Period. Polytech. Transp. Eng.* **2016**, *44*, 244–250. [CrossRef]
2. Mihály, A.; Farkas, Z.; Gáspár, P. Multicriteria Autonomous Vehicle Control at Non-Signalized Intersections. *Appl. Sci.* **2020**, *10*, 7161. [CrossRef]
3. Van Brummelen, J.; O'Brien, M.; Gruyer, D.; Najjaran, H. Autonomous vehicle perception: The technology of today and tomorrow. *Transp. Res. Part C Emerg. Technol.* **2018**, *89*, 384–406. [CrossRef]
4. Huang, W.; Wang, K.; Lv, Y.; Zhu, F. Autonomous vehicles testing methods review. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 163–168.
5. Leneman, F.; Verburg, D.; Buijssen, S. PreScan, testing and developing active safety applications through simulation. In Proceedings of the 3. Tagung Aktive Sicherheit durch Fahrerassistenz, München, Germany, 7–8 April 2008.
6. Tsogas, M.; Polychronopoulos, A.; Amditis, A. Unscented Kalman filter design for curvilinear motion models suitable for automotive safety applications. In Proceedings of the 2005 7th International Conference on Information Fusion, Philadelphia, PA, USA, 25–28 July 2005; Volume 2.
7. Schubert, R.; Richter, E.; Wanielik, G. Comparison and evaluation of advanced motion models for vehicle tracking. In Proceedings of the 2008 11th International Conference on Information Fusion, Cologne, Germany, 30 June–3 July 2008; pp. 1–6.
8. Sarkar, S.; Roy, A. Interacting Multiple Model (IMM) algorithm for road object tracking using automotive radar. In Proceedings of the 11th International Radar Symposium India, Bangalore, India, 12–16 December 2017.
9. Bar-Shalom, Y.; Li, X.R.; Kirubarajan, T. *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*; John Wiley & Sons: Hoboken, NJ, USA, 2004.
10. Blom, H.A.; Bar-Shalom, Y. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Trans. Autom. Control* **1988**, *33*, 780–783. [CrossRef]
11. Wenkang, W.; Jingan, F.; Bao, S.; Xinxin, L. Vehicle State Estimation Using Interacting Multiple Model Based on Square Root Cubature Kalman Filter. *Appl. Sci.* **2021**, *11*, 10772. [CrossRef]
12. Mazor, E.; Averbuch, A.; Bar-Shalom, Y.; Dayan, J. Interacting multiple model methods in target tracking: A survey. *IEEE Trans. Aerosp. Electron. Syst.* **1998**, *34*, 103–123. [CrossRef]
13. Liu, Z.; Wang, J. Interacting multiple model gaussian particle filter. In Proceedings of the 2011 9th World Congress on Intelligent Control and Automation (WCICA), Taipei, Taiwan, 21–25 June 2011; pp. 270–273.

14. Du, S.c.; Shi, Z.g.; Zang, W.; Chen, K.s. Using interacting multiple model particle filter to track airborne targets hidden in blind Doppler. *J. Zhejiang Univ.-Sci. A* **2007**, *8*, 1277–1282. [CrossRef]
15. Wang, X.; Xu, M.; Wang, H.; Wu, Y.; Shi, H. Combination of interacting multiple models with the particle filter for three-dimensional target tracking in underwater wireless sensor networks. *Math. Probl. Eng.* **2012**, *2012*, 829451. [CrossRef]
16. Guo, R.; Qin, Z.; Li, X.; Chen, J. An IMMUPF method for ground target tracking. In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Toronto, ON, Canada, 11–14 October 2007; pp. 96–101.
17. Foo, P.H.; Ng, G.W. Combining IMM method with particle filters for 3D maneuvering target tracking. In Proceedings of the 2007 10th International Conference on Information Fusion, Québec, QC, Canada, 9–12 July 2007; pp. 1–8.
18. Törő, O.; Bécsi, T.; Aradi, S.; Gáspár, P. IMM Bernoulli Gaussian Particle Filter. *IFAC-PapersOnLine* **2018**, *51*, 274–279. [CrossRef]
19. Gupta, N.; Hauser, R. Kalman filtering with equality and inequality state constraints. *arXiv* **2007**, arXiv:0709.2791.
20. Fang, Y.; Wang, C.; Yao, W.; Zhao, X.; Zhao, H.; Zha, H. On-road vehicle tracking using part-based particle filter. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 4538–4552. [CrossRef]
21. Gordon, N.; Ristic, B.; Arulampalam, S. Beyond the kalman filter: Particle filters for tracking applications. *Artech House Lond.* **2004**, *830*, 1–4.
22. Kowalczuk, Z.; Sankowski, M. Soft- and Hard-Decision Multiple-Model Estimators for Air Traffic Control. *IEEE Trans. Aerosp. Electron. Syst.* **2010**, *46*, 2056–2065. [CrossRef]
23. Farmer, M.E.; Hsu, R.L.; Jain, A.K. Interacting multiple model (IMM) Kalman filters for robust high speed human motion tracking. In Proceedings of the Object Recognition Supported by User Interaction for Service Robots, Quebec City, QC, Canada, 11–15 August 2002; Volume 2, pp. 20–23.
24. Zheng, W.; Bhandarkar, S.M. Face detection and tracking using a boosted adaptive particle filter. *J. Vis. Commun. Image Represent.* **2009**, *20*, 9–27. [CrossRef]
25. Kim, T.; Park, T.H. Extended Kalman filter (EKF) design for vehicle position tracking using reliability function of radar and lidar. *Sensors* **2020**, *20*, 4126. [CrossRef]
26. Zhao, D.; Fu, H.; Xiao, L.; Wu, T.; Dai, B. Multi-object tracking with correlation filter for autonomous vehicle. *Sensors* **2018**, *18*, 2004. [CrossRef] [PubMed]
27. Rákos, O.; Aradi, S.; Bécsi, T. Lane Change Prediction Using Gaussian Classification, Support Vector Classification and Neural Network Classifiers. *Period. Polytech. Transp. Eng.* **2020**, *48*, 327–333. [CrossRef]
28. Törő, O.; Bécsi, T.; Gáspár, P. PHD Filter for Object Tracking in Road Traffic Applications Considering Varying Detectability. *Sensors* **2021**, *21*, 472. [CrossRef] [PubMed]
29. Ormsby, C.D.; Raquet, J.F.; Maybeck, P.S. A new generalized residual multiple model adaptive estimator of parameters and states. *Math. Comput. Model.* **2006**, *43*, 1092–1113. [CrossRef]
30. Henderson, P.E.; Raquet, J.F.; Maybeck, P.S. A Multiple Filter Approach for Precise Kinematic DGPS Positioning and Carrier-Phase Ambiguity Resolution. *Navigation* **2002**, *49*, 149–160. [CrossRef]
31. Zhang, L.; Sidoti, D.; Bienkowski, A.; Pattipati, K.R.; Bar-Shalom, Y.; Kleinman, D.L. On the Identification of Noise Covariances and Adaptive Kalman Filtering: A New Look at a 50 Year-old Problem. *IEEE Access* **2020**, *8*, 59362–59388. [CrossRef] [PubMed]
32. Maybeck, P.S. *The Kalman Filter: An Introduction to Concepts*; Springer: New York, NY, USA, 1990.
33. Simon, D. Kalman filtering with state constraints: A survey of linear and nonlinear algorithms. *IET Control Theory Appl.* **2010**, *4*, 1303–1318. [CrossRef]
34. Teixeira, B.O.S.; Chandrasekar, J.; Torres, L.A.; Aguirre, L.A.; Bernstein, D.S. State estimation for equality-constrained linear systems. In Proceedings of the 2007 46th IEEE Conference on Decision and Control, New Orleans, LA, USA, 12–14 December 2007; pp. 6220–6225.
35. Li, X.R.; Jilkov, V.P. Survey of maneuvering target tracking. Part V. Multiple-model methods. *IEEE Trans. Aerosp. Electron. Syst.* **2005**, *41*, 1255–1321.
36. Granström, K.; Willett, P.; Bar-Shalom, Y. Systematic approach to IMM mixing for unequal dimension states. *IEEE Trans. Aerosp. Electron. Syst.* **2015**, *51*, 2975–2986. [CrossRef]
37. Boers, Y.; Driessen, H.; Bagchi, A. Point estimation for jump Markov systems: Various MAP estimators. In Proceedings of the 2009 12th International Conference on Information Fusion, Seattle, WA, USA, 6–9 July 2009; pp. 33–40.
38. Yu, C.H.; Zhuang, H.Q.; Seo, T.I.; Kim, E.J. VSIMM Based Target Tracking Filter Design. In Proceedings of the 2009 Korea Automatic Control Conference, Jeju Island, Korea, 10–12 December 2009; pp. 76–80.
39. Baxter, R.H.; Leach, M.J.; Mukherjee, S.S.; Robertson, N.M. An adaptive motion model for person tracking with instantaneous head-pose features. *IEEE Signal Process. Lett.* **2014**, *22*, 578–582. [CrossRef]