

Article

Hybrid Interacting Multiple Model Filtering for Improving the Reliability of Radar-Based Forward Collision Warning Systems

Jung Min Pak 

Department of Electrical Engineering, Wonkwang University, 460 Iksan-Daero, Iksan 54538, Korea;
destin11@wku.ac.kr

Abstract: Automotive forward collision warning (FCW) systems based on radar sensors attracted widespread attention in recent years. To achieve a reliable FCW, it is essential to accurately estimate the position and velocity of a preceding vehicle. To this end, this study proposed a novel estimation algorithm, which is a hybrid of interacting multiple model probabilistic data association (IMM-PDA) and finite impulse response (FIR) filters. Although the IMM-PDA filter is one of the most successful algorithm for tracking a maneuvering target in clutters, it sometimes exhibits divergence owing to modeling errors. In this study, the divergent IMM-PDA filter in the novel algorithm was reset and recovered using an assisting FIR filter. Consequently, this enabled reliable estimation for FCW. The improved reliability of the proposed algorithm was demonstrated through the simulation of preceding vehicle tracking using automotive radars.

Keywords: automotive radar; finite impulse response filter; forward collision warning; interacting multiple model; probabilistic data association



Citation: Pak, J.M. Hybrid Interacting Multiple Model Filtering for Improving the Reliability of Radar-Based Forward Collision Warning Systems. *Sensors* **2022**, *22*, 875. <https://doi.org/10.3390/s22030875>

Academic Editors: Lei Zhang and Yulai Cong

Received: 23 December 2021

Accepted: 21 January 2022

Published: 24 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, advanced driver-assistance systems (ADAS) were developed and attracted widespread attention [1]. Among various ADAS, forward collision warning (FCW) systems are essential for the safety of drivers [2]. FCW can be implemented using various sensors, such as radar [3], vision [4], and global positioning system (GPS) [5]. Radar-based FCW is the most commonly used FCW owing to the fact that radars can provide long-range detection regardless of weather/light conditions [6–8].

Studies on FCW algorithms focused on the decision-making process involved in determining whether to alert a collision. In typical FCW systems [9–12], the time-to-collision (TTC) is computed using the information of the relative distance and velocity between the host and preceding vehicles, and a warning is provided if the TTC was below a certain threshold. Recent studies [13–15] analyzed the behaviors (driving patterns) of vehicles using artificial intelligence (AI) algorithms, and a warning is provided in situations with a high risk of collision. Although various decision-making algorithms for FCW were developed, most of them are based on information on the state (position and velocity) of preceding vehicles. Thus, it is essential to develop algorithms that can accurately estimate the state of a preceding vehicle to achieve a reliable FCW [5,9,11].

The estimation of the position and velocity of a target, known as target tracking, requires two processes: signal processing and data processing. In radar signal processing, measurements of the range, bearing, and range rate are obtained by processing the received radar signals, and this can be achieved using various signal processing methods including Fourier analysis, filtering, and AI algorithms. The measurements obtained using signal processing contain noises and clutters; hence, a postprocessing, known as data processing, is required. Stochastic filters (e.g., Kalman filter (KF)) are used in data processing to improve the accuracy of the estimation of the position and velocity of a target. In addition, data association algorithms (e.g., probabilistic data association) are used to

address the problem of clutters (i.e., false measurements). Probabilistic data association filter (PDAF) [16], which is the combination of KF and probabilistic data association, is one of the most renowned and successful algorithms for single target tracking in the presence of clutters. Particularly, interacting multiple model PDAF (IMMPDAF) [17–20] is an advanced algorithm for maneuvering targets, and joint PDAF [21] is an algorithm for multiple target tracking. Recently, various multiple target tracking algorithms, such as multiple hypothesis tracking [22] and probability hypothesis density filter [23], were developed. This study focused on data processing, particularly, the problem of single vehicle tracking in the presence of clutters, for application in FCW systems. Thus, IMMPDAF was used as the main filter in the proposed algorithm.

However, as the IMMPDAF is based on the KF, it sometimes exhibits estimation divergence owing to modeling errors. For example, a significant difference in the actual movement of a target and the movement predicted by the motion model may result in the divergence of KF. This divergence could be attributed to the accumulation of errors in the KF. To overcome the filter divergence problem, stochastic filters with finite impulse response (FIR) structures were investigated and developed [24–31]. FIR stochastic filters, known as FIR filters, only utilize recent finite measurements to produce a current state estimate, which makes them robust against modeling errors. However, because they utilize finite measurements, the accuracy of their estimates are generally lower than that of KF that utilizes all past measurements. In summary, the IMMPDAF is accurate under normal conditions, but may diverge if the actual movement of a target is significantly different from that of the motion model. In contrast, FIR filters do not diverge, but are less accurate under normal conditions.

Therefore, this study proposed a novel hybrid estimation algorithm for improving the reliability of radar-based FCW. The proposed algorithm utilized IMMPDAF as the main filter, and an FIR filter as an assisting filter. The assisting FIR filter only function when the IMMPDAF diverges, and it produces a state estimate and covariance. In this situation, the IMMPDAF is reset using the information obtained from the FIR filter. Consequently, the IMMPDAF recovers from failures. Therefore, the proposed algorithm was named hybrid IMM-PDA/FIR filter (HIPFF). The performance of the proposed HIPFF was demonstrated using simulations, and the relative position and velocity of a vehicle were estimated using the HIPFF and conventional IMMPDAF. The results revealed that when a preceding vehicle decelerated rapidly, the conventional IMMPDAF often failed and diverged, whereas the proposed HIPFF did not. This indicates that HIPFF significantly improved the reliability of the state estimation for FCW systems.

The rest of this paper is organized as follows. Section 2 introduces the problem involved in the estimation of the position and velocity of a preceding vehicle in the presence of radar clutters. Section 3 introduces the IMMPDAF algorithm, and Section 4 introduces the proposed HIPFF algorithm. Section 5 presents the simulation results to demonstrate the performance of the proposed HIPFF, and the conclusions are drawn in Section 6.

2. Preceding Vehicle State Estimation Using Automotive Radars

This section describes the problem involved in the estimation of the state of a preceding vehicle using an automotive radar in the presence of clutters. The state (position and velocity) of the preceding vehicle is represented using a two-dimensional (2D) Cartesian coordinate system (Figure 1). An automotive radar for FCW is installed in the middle of the host vehicle, and the origin of the coordinate system is located at the position of the radar. The x-axis is aligned with the longitudinal direction of the host vehicle. The state of a preceding vehicle (target) is composed of 2D positions, (x, y) , and velocities, (v_x, v_y) . Consequently, the state vector is defined as $\mathbf{x} = [x \ v_x \ y \ v_y]^T$.

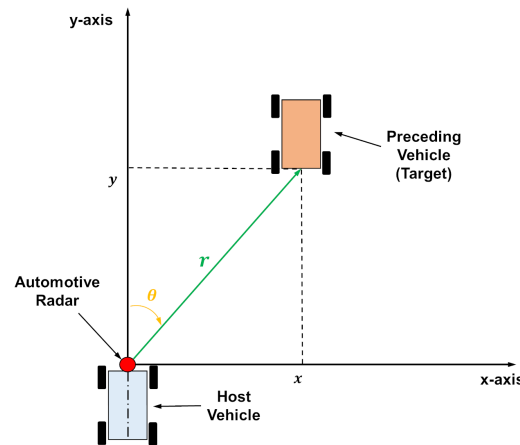


Figure 1. Automotive radar geometry.

Estimation algorithms require state-space models, which consist of motion and measurement models. The motion model expresses the transition of the state vector in discrete-time. Typically, the motion model used for target tracking is the discrete white noise acceleration (DWNA) model, which can be represented using:

$$\mathbf{x}(k+1) = \mathbf{F}\mathbf{x}(k) + \mathbf{G}\mathbf{w}(k),$$

$$\mathbf{F} = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{bmatrix}, \quad (1)$$

where $\mathbf{x}(k)$ is the state vector at a discrete time k and $\mathbf{w}(k)$ is the zero-mean white Gaussian process noise vector. The covariance of $\mathbf{w}(k)$ is defined as $\mathbf{Q}(k) = \sigma_w^2 \mathbf{I}_2$, where σ_w^2 is the variance of process noise, and \mathbf{I}_2 is the 2×2 identity matrix.

The process noise, $\mathbf{w}(k)$, corresponds to the acceleration of a target, and σ_w^2 determines the level of acceleration. This indicates that σ_w^2 reflects the amount of changes in velocity. When an appropriate value of σ_w^2 is used, the filter can produce accurate state estimates. However, if σ_w^2 is not sufficiently large, and there is a significant change in the velocity of a target, the filter divergence phenomenon (i.e., rapid increase in estimation error) may occur. In contrast, if σ_w^2 was sufficiently large and the changes in velocity was too small, the estimation accuracy is degraded. Thus, a mismatch between the model (i.e., σ_w^2) and the actual motion of a target can result in a degradation in the estimation accuracy or even filter divergence. However, the acceleration of a target is generally unknown, and σ_w^2 is determined using a rule of thumb based on the knowledge of engineers. The IMM estimation algorithm can solve the problem of determining the uncertain σ_w^2 . The IMM algorithm utilizes multiple models that use different σ_w^2 . A detailed algorithm for the IMM estimation is described in Section 3.

The radar reports the relative range, r , and bearing, θ , between the host and preceding vehicle at a constant time interval, T , as shown in Figure 1. These radar measurements including noise can be expressed as

$$\begin{aligned} r_m &= r + \bar{v}_r, \\ \theta_m &= \theta + \bar{v}_\theta, \end{aligned} \quad (2)$$

where r and θ are the true range and bearing, respectively; r_m and θ_m are the measured range and bearing, respectively; and \bar{v}_r and \bar{v}_θ are the measurement noises, which are assumed to be mutually independent and zero-mean white Gaussian. The variances of \bar{v}_r and \bar{v}_θ are denoted as σ_r^2 and σ_θ^2 , respectively.

The measurement model relates the radar measurements with the state, $\mathbf{x}(k)$. However, the raw radar measurements are polar coordinates, and they should be converted to Cartesian coordinates. The conversion can be written as

$$\begin{aligned}x_m &= r_m \cos(\theta_m), \\y_m &= r_m \sin(\theta_m),\end{aligned}\quad (3)$$

where x_m and y_m are the converted measurements, and they can also be represented as

$$\begin{aligned}x_m &= x + \bar{v}_x, \\y_m &= y + \bar{v}_y,\end{aligned}\quad (4)$$

where \bar{v}_x and \bar{v}_y are the measurement noises in the Cartesian coordinate.

The measurement vector is defined as $\mathbf{z}(k) = [x_m(k) \ y_m(k)]^T$, and the measurement model is represented as follows:

$$\begin{aligned}\mathbf{z}(k) &= \mathbf{H}\mathbf{x}(k) + \bar{\mathbf{v}}(k), \\ \mathbf{H} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},\end{aligned}\quad (5)$$

where $\bar{\mathbf{v}}(k)$ is the measurement noise vector. The noise covariance of converted measurement, denoted as $\mathbf{R}(k)$, is obtained as follows [17]:

$$\begin{aligned}\mathbf{R}(k) &= \begin{bmatrix} R_{11}(k) & R_{12}(k) \\ R_{12}(k) & R_{22}(k) \end{bmatrix}, \\ R_{11}(k) &= r_m(k)^2 \sigma_\theta^2 \sin^2(\theta_m(k)) + \sigma_r^2 \cos^2(\theta_m(k)), \\ R_{22}(k) &= r_m(k)^2 \sigma_\theta^2 \cos^2(\theta_m(k)) + \sigma_r^2 \sin^2(\theta_m(k)), \\ R_{12}(k) &= (\sigma_r^2 - r_m(k)^2 \sigma_\theta^2) \sin(\theta_m(k)) \cos(\theta_m(k)).\end{aligned}\quad (6)$$

Given the motion and measurement models, Equations (1) and (5), the state $\mathbf{x}(k)$ can be estimated using various estimation algorithms. However, two factors hinder this estimation: radar clutters and maneuvering of target. Radar clutters are false measurements generated by the scattering of radar reflection. Because it is unknown whether a measurement is true or false, false measurements make the estimation difficult. Nevertheless, this uncertainty in measurements can be handled by PDA algorithm. Maneuvering corresponds to changes in the direction of movement and speed. A rapid and significant maneuvering of a target results in a mismatch between the motion model (particularly σ_w^2) and the actual motion of the target, which can result in divergence of the estimation algorithm. This uncertainty in the motion model can be handled by IMM algorithm. In the following section, IMMPDAF, which is a combination of IMM and PDA algorithms, is introduced.

3. Interacting Multiple Model (IMM) Probabilistic Data Association (PDA) Filter

The IMMPDAF operates multiple PDA filters in parallel, and each PDA filter is matched to a model. The inputs and outputs of the multiple PDA filters were mixed using the IMM algorithm. In this section, first, the PDA filter is introduced, after which the IMM algorithm is described.

The PDA filter is a recursive estimator, indicating that the processes are repeatedly performed at each time step. The first process of the PDA filter is a time update process, in which the state estimate and estimation error covariance are updated using the motion model as follows:

$$\begin{aligned}\hat{\mathbf{x}}(k|k-1) &= \mathbf{F}\hat{\mathbf{x}}(k-1|k-1), \\ \mathbf{P}(k|k-1) &= \mathbf{F}\mathbf{P}(k-1|k-1)\mathbf{F}^T + \mathbf{G}\mathbf{Q}(k)\mathbf{G}^T.\end{aligned}\quad (7)$$

where the notation “ $k|k-1$ ” corresponds to *a priori* estimates obtained at time k using the measurements up to time $k-1$. In addition, the notation “ $k-1|k-1$ ” corresponds to *a posteriori* estimates obtained at $k-1$ using the measurements up to $k-1$.

The second process is measurement validation, in which only “valid” measurements are selected. The number of measurements obtained at time k is denoted as $m(k)$, and i -th measurement ($i = 1, 2, \dots, m(k)$) at time k is denoted as $\mathbf{z}_i(k)$. The measurements within the validation region (also known as the gate) are selected as follows:

$$\begin{aligned} v_i(k) &= (\mathbf{z}_i(k) - \hat{\mathbf{z}}(k))^T \mathbf{S}_i^{-1}(k) (\mathbf{z}_i(k) - \hat{\mathbf{z}}(k)) \leq \gamma, \\ \hat{\mathbf{z}}(k) &= \mathbf{H} \hat{\mathbf{x}}(k|k-1), \\ \mathbf{S}_i(k) &= \mathbf{H} \mathbf{P}(k|k-1) \mathbf{H}^T + \mathbf{R}_i(k), \end{aligned} \quad (8)$$

where γ is the gate threshold, which can be obtained from a *chi-square* table, and $\mathbf{R}_i(k)$ is the measurement noise covariance computed using $\mathbf{z}_i(k)$ and (6). The number of selected (validated) measurements is denoted as $n(k)$.

The third process is the computation of association probability. The association probability, $\beta_i(k)$, is the probability that the measurement $\mathbf{z}_i(k)$ is from the true target (i.e., true measurement), which can be computed as follows:

$$\beta_i(k) = \begin{cases} \frac{\mathcal{L}_i(k)}{1 - P_D P_G + \sum_{j=1}^{n(k)} \mathcal{L}_j(k)} & i = 1, \dots, n(k) \\ \frac{1 - P_D P_G}{1 - P_D P_G + \sum_{j=1}^{n(k)} \mathcal{L}_j(k)} & i = 0 \end{cases} \quad (9)$$

$$\mathcal{L}_j(k) = \frac{\exp[-0.5(\mathbf{z}_j(k) - \hat{\mathbf{z}}(k))^T \mathbf{S}_j^{-1}(k) (\mathbf{z}_j(k) - \hat{\mathbf{z}}(k))]}{(2\pi)^{n_z/2} |\mathbf{R}_j(k)|^{1/2} n(k)} P_D V_j(k) \quad (10)$$

$$V_j(k) = c_{n_z} \gamma^{n_z/2} |\mathbf{S}_j(k)|^{1/2}, \quad (11)$$

where $i = 0$ corresponds to the case where there is no true measurement, and $V_j(k)$ is the volume of the validation region, which is computed as

$$V_k(j) = c_{n_z} \gamma^{n_z/2} |\mathbf{S}_k(j)|^{1/2}, \quad (12)$$

where n_z is the dimension of the measurement vector. In addition, P_G and P_D in (9) are the gate probability and the target detection probability, respectively.

The last process is the measurement update, in which state estimates are updated using the validated measurements as follows:

$$\begin{aligned} \hat{\mathbf{x}}(k|k) &= \hat{\mathbf{x}}(k|k-1) + \sum_{i=1}^{n(k)} \mathbf{W}_i(k) \beta_i(k) v_i(k), \\ \mathbf{W}_i(k) &= \mathbf{P}(k|k-1) \mathbf{H}^T \mathbf{S}_i^{-1}(k), \\ v_i(k) &= \mathbf{z}_i(k) - \hat{\mathbf{z}}(k), \\ \mathbf{P}(k|k) &= \bar{\mathbf{P}}(k|k) + \tilde{\mathbf{P}}(k|k), \\ \bar{\mathbf{P}}(k|k) &= \beta_0(k) \mathbf{P}(k|k-1) + \sum_{i=1}^{n(k)} \beta_i(k) (\mathbf{I} - \mathbf{W}_i(k) \mathbf{H}) \mathbf{P}(k|k-1), \\ \tilde{\mathbf{P}}(k|k) &= \sum_{i=1}^{n(k)} \beta_i(k) \mathbf{W}_i(k) v_i(k) v_i^T(k) \mathbf{W}_i^T(k) \\ &\quad - \left(\sum_{i=1}^{n(k)} \beta_i(k) \mathbf{W}_i(k) v_i(k) \right) \left(\sum_{i=1}^{n(k)} \beta_i(k) v_i^T(k) \mathbf{W}_i^T(k) \right). \end{aligned} \quad (13)$$

If there is no measurement in the validation region (i.e., $n(k) = 0$), the measurement update process is omitted.

Next, the IMM algorithm is introduced. In this study, the IMM algorithm uses the following models:

- M_1 : DWNA model (1) assuming undetected target (false target)
- M_2 : DWNA model (1) assuming true target with normal motion
- M_3 : DWNA model (1) assuming true target performing maneuvering
- M_4 : DWNA model (1) assuming true target with nearly constant velocity

The difference between M_1 and M_2 is the target detection probability, which can be expressed as follows:

$$P_D^j = \begin{cases} 0, & \text{if } j = 1, \\ P_D, & \text{if } j = 2, 3, 4. \end{cases} \quad (14)$$

where j is the model (mode) number. The model M_1 plays a role in detecting a target loss. The differences among M_2 – M_4 are the process noise variance σ_w^2 . M_3 uses the highest σ_w^2 , and M_4 uses the lowest σ_w^2 . A PDA filter that uses model j is denoted as filter j ($j = 1, 2, 3, 4$).

First, IMM-PDA algorithm computes the mixing probability as

$$\begin{aligned} \mu_{h|j}(k-1|k-1) &= \frac{1}{\bar{c}_j} p_{hj} \mu_h(k-1), \\ \bar{c}_j &= \sum_{h=1}^r p_{hj} \mu_h(k-1), \end{aligned} \quad (15)$$

where $\mu_{h|j}(k-1|k-1)$ is the mixing probability, indicating the weight imposed on the information from filters h to i .

Using the mixing probabilities, initial estimate and covariance for the filter j are obtained as follows:

$$\hat{\mathbf{x}}^{0j}(k-1|k-1) = \sum_{h=1}^4 \hat{\mathbf{x}}^{0j}(k-1|k-1) \mu_{h|j}(k-1|k-1), \quad (16)$$

$$\begin{aligned} \mathbf{P}^{0j}(k-1|k-1) &= \sum_{h=1}^4 \mu_{h|j}(k-1|k-1) \{ \mathbf{P}^h(k-1|k-1) + [\hat{\mathbf{x}}^h(k-1|k-1) \\ &\quad - \hat{\mathbf{x}}^{0j}(k-1|k-1)][\hat{\mathbf{x}}^h(k-1|k-1) - \hat{\mathbf{x}}^{0j}(k-1|k-1)]^T \}. \end{aligned} \quad (17)$$

$\hat{\mathbf{x}}^{0j}(k-1|k-1)$ and $\mathbf{P}^{0j}(k-1|k-1)$ are fed to the filter j , which corresponds to $\hat{\mathbf{x}}(k-1|k-1)$ and $\mathbf{P}(k-1|k-1)$ in (7), respectively. Subsequently, PDA filtering (7)–(13) is performed for each model j , and $\hat{\mathbf{x}}^j(k|k)$ and $\mathbf{P}^j(k|k)$ are obtained

Next, the likelihoods for $\hat{\mathbf{x}}^j(k|k)$ are computed as follows:

$$\Lambda_j[Z(k)] = \frac{1 - P_D^j}{V^{n(k)}} + \frac{n(k) P_D^j}{V^{n(k)-1}} \sum_{i=1}^{n(k)} \mathcal{N}[\mathbf{z}_i(k); \hat{\mathbf{z}}^j(k|k-1), \mathbf{S}^j(k)], \quad (18)$$

$$V = \pi \gamma |\mathbf{S}^j(k)|^{1/2}, \quad (19)$$

$$\mathbf{S}^j(k) = \mathbf{H} \mathbf{P}^j(k|k) \mathbf{H}^T + \mathbf{R}_i(k), \quad (20)$$

$$\hat{\mathbf{z}}^j(k|k-1) = \mathbf{H} \hat{\mathbf{x}}^j(k|k), \quad (21)$$

where $\mathcal{N}[\mathbf{z}_i(k); \hat{\mathbf{z}}^j(k|k-1), \mathbf{S}^j(k)]$ corresponds to the Gaussian probability density function with the mean $\hat{\mathbf{z}}^j(k|k-1)$ and covariance $\mathbf{S}^j(k)$; $Z(k)$ is the set of validated measurements defined as $Z(k) = \{\mathbf{z}_i(k)\}_{i=1}^{n(k)}$. Using the likelihoods, mode probabilities are updated as

$$\mu_j(k) = \frac{1}{c} \Lambda_j[Z(k)] \bar{c}_j, \quad (22)$$

$$c = \sum_{j=1}^4 \Lambda_j[Z(k)] \bar{c}_j. \quad (23)$$

Lastly, the ultimate estimate and covariance at time k are obtained by combination as follows:

$$\hat{\mathbf{x}}(k|k) = \sum_{j=1}^4 \hat{\mathbf{x}}^j(k|k) \mu_j(k), \quad (24)$$

$$\mathbf{P}(k|k) = \sum_{j=1}^4 \mu_j(k) \{ \mathbf{P}^j(k|k) + [\hat{\mathbf{x}}^j(k|k) - \hat{\mathbf{x}}(k|k)][\hat{\mathbf{x}}^j(k|k) - \hat{\mathbf{x}}(k|k)]^T \}. \quad (25)$$

At the initial time of the IMMPDAF filtering, the mode probabilities were set equally as $\mu_j(k) = 0.25$. The transition probabilities from mode h to j , which is denoted as p_{hj} , is an important design parameter that affects the performance of the IMMPDAF algorithm. If the characteristics of the target motion are known, p_{hj} can be designed using the information on the mean sojourn time for each mode. However, if the target motion information is not available, p_{hj} can be set equally.

The IMMPDAF can handle the problem involved in the selection of a value for uncertain σ_w^2 in the motion model by adopting multiple models that use different σ_w^2 values. However, the IMMPDAF exhibits certain limitations. First, it has a limited number of models because of computational burden. Second, the multiple models are selected based on the knowledge of engineers, and inappropriate models are sometimes selected. In target tracking problems, even IMMPDAF can fail and diverge if the changes in target velocity are significantly different from the σ_w^2 value in the motion model (e.g., rapid deceleration of vehicles owing to full braking). Thus, this study proposed a novel hybrid filtering algorithm that can recover the IMMPDAF from failures. Failure of IMMPDAF under severe conditions and recovering the failed algorithm were demonstrated using simulation results.

4. Hybrid IMM-PDA/Finite Impulse Response (FIR) Filter

In this section, the proposed HIPFF algorithm is introduced. In the HIPFF algorithm, the IMMPDAF acts as the main filter, and a FIR filter was used to assist the main filter. This assisting FIR filter operates when the IMMPDAF algorithm fails. Thus, the first problem is to detect the failures of the IMMPDAF. In this study, target loss indicates that the estimated position is far away from the actual target position. If the IMMPDAF loses the target, M_1 exhibits the highest mode (model) probability among the four models. The mode probability of M_1 is denoted as $\mu_1(k)$. Thus, a target loss can be detected using $\mu_1(k)$. However, it is too late to operate the assisting FIR filter after the detection of a target loss. If there are no measurements in the validation region ($n(k) = 0$), the IMMPDAF cannot perform the measurement update, which may result in a target loss. Thus, $n(k) = 0$ can be considered as a symptom of failures. The proposed HIPFF algorithm operated the assisting FIR filter when $n(k) = 0$, and obtained the state estimate and covariance for resetting the main filter. A minimum variance FIR filter (MVFF) was adopted for the HIPFF algorithm. The MVFF is a one-step prediction estimator, and it can produce state estimate and covariance at the current time k using the measurements on the horizon $k - N \leq k \leq k - 1$, where N is the horizon size of the FIR filter. Moreover, because the MVFF does not use the current measurement, it is suitable for the employed assisting

filter when there is no validated measurements. The equations of the MVFF can be written as follows:

$$\hat{\mathbf{x}}^*(k|k-1) = \mathbf{L} \mathbf{Z}_N, \quad (26)$$

$$\mathbf{P}^*(k|k-1) = \mathbf{K}_N \mathbf{Q}_N \mathbf{K}_N^T + \mathbf{L} \mathbf{R}_N \mathbf{L}^T, \quad (27)$$

$$\mathbf{L} = \mathbf{J}_N \begin{bmatrix} \mathbf{M}_{1,1} & \mathbf{M}_{1,2} \\ \mathbf{M}_{1,2}^T & \mathbf{M}_{2,2} \end{bmatrix}^{-1} \begin{bmatrix} \bar{\mathbf{C}}_N^T \\ \bar{\mathbf{G}}_N^T \end{bmatrix} \mathbf{R}_N^{-1}, \quad (28)$$

$$\mathbf{J}_N = [\mathbf{F}^N \ \mathbf{F}^{N-1} \ \mathbf{F}^{N-2} \ \dots \ \mathbf{F} \ \mathbf{I}], \quad (29)$$

$$\mathbf{M}_{1,1} = \bar{\mathbf{C}}_N^T \mathbf{R}_N^{-1} \bar{\mathbf{C}}_N,$$

$$\mathbf{M}_{1,2} = \bar{\mathbf{C}}_N^T \mathbf{R}_N^{-1} \bar{\mathbf{G}}_N,$$

$$\mathbf{M}_{2,2} = \bar{\mathbf{G}}_N^T \mathbf{R}_N^{-1} \bar{\mathbf{G}}_N + \mathbf{Q}_N^{-1}, \quad (30)$$

$$\bar{\mathbf{C}}_N = \begin{bmatrix} \mathbf{H} \\ \mathbf{HF} \\ \mathbf{HF}^2 \\ \vdots \\ \mathbf{HF}^{N-1} \end{bmatrix}, \quad (31)$$

$$\bar{\mathbf{G}}_N = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{HG} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{HFG} & \mathbf{HG} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{HF}^{N-1}\mathbf{G} & \mathbf{HF}^{N-2}\mathbf{G} & \dots & \mathbf{HG} & \mathbf{0} \end{bmatrix}, \quad (32)$$

$$\mathbf{R}_N = [\text{diag}(\overbrace{\mathbf{R}^*(k-N) \ \mathbf{R}^*(k-N+1) \ \dots \ \mathbf{R}^*(k-1)}^N)], \quad (33)$$

$$\mathbf{Q}_N = [\text{diag}(\overbrace{\mathbf{Q}^*(k-N) \ \mathbf{Q}^*(k-N+1) \ \dots \ \mathbf{Q}^*(k-1)}^N)], \quad (34)$$

$$\mathbf{Z}_N = [\bar{\mathbf{z}}^T(k-N) \ \bar{\mathbf{z}}^T(k-N+1) \ \dots \ \bar{\mathbf{z}}^T(k-1)]^T, \quad (35)$$

$$\mathbf{K}_N = [\mathbf{F}^{N-1}\mathbf{G} \ \mathbf{F}^{N-2}\mathbf{G} \ \dots \ \mathbf{FG} \ \mathbf{G}]. \quad (36)$$

Equations (26) and (35) indicate that the MVFF cannot handle clutters, and the most suitable measurement should be selected at each time step on the horizon $[k-N, k-1]$. The most suitable measurement can be selected using the nearest neighbor method as follows:

$$\begin{aligned} \bar{\mathbf{z}}(l) &= \arg \min_{\mathbf{z}_i(l)} (\mathbf{z}_i(l) - \hat{\mathbf{z}}(l))^T \mathbf{S}_i^{-1}(l) (\mathbf{z}_i(l) - \hat{\mathbf{z}}(l)), \quad i = 1, \dots, m(l), \\ \hat{\mathbf{z}}(l) &= \mathbf{H} \hat{\mathbf{x}}(l|l), \\ \mathbf{S}_i(l) &= \mathbf{H} \mathbf{P}(l|l) \mathbf{H}^T + \mathbf{R}_i(l), \end{aligned} \quad (37)$$

where l is the time on the horizon $[k-N, k-1]$.

Equations (33) and (34) indicate that the MVFF can handle time-varying noise covariances. In (33), $\mathbf{R}^*(l)$ ($l = k-N, \dots, k-1$) can be obtained using $\bar{\mathbf{z}}(l)$ and (6). To this end, $\bar{\mathbf{z}}$ can be converted into a polar coordinate using the relation as follows:

$$\begin{aligned} r_m &= \sqrt{x_m^2 + y_m^2}, \\ \theta_m &= \tan^{-1}(\frac{y_m}{x_m}). \end{aligned} \quad (38)$$

In (34), $\mathbf{Q}^*(l)$ ($l = k-N, \dots, k-1$) is determined by selecting one of the process noise covariances of the four models (M_1 – M_4). The process noise covariance of the model producing the highest mode probability is selected as $\mathbf{Q}^*(l)$.

After the outputs of the MVFF, $\hat{\mathbf{x}}^*(k|k-1)$ and $\mathbf{P}^*(k|k-1)$, are obtained, all the PDA filters in the main filter are reset using:

$$\hat{\mathbf{x}}^j(k|k) = \hat{\mathbf{x}}^*(k|k-1), \quad (39)$$

$$\mathbf{P}^j(k|k) = \mathbf{P}^*(k|k-1), \quad (j = 1, 2, 3, 4). \quad (40)$$

The overall HIPFF algorithm is summarized in Algorithm 1.

Algorithm 1: HIPFF

Data: $\mathbf{Z}(k), \bar{\mathbf{z}}(l), \mathbf{R}^*(l), \mathbf{Q}^*(l)$ ($l = k - N, \dots, k - 1$)

Result: $\hat{\mathbf{x}}(k|k), \mathbf{P}(k|k)$

```

1 begin
2   - Compute the mixing probabilities using (15).
3   - Obtain an initial estimate and a covariance for the model-matched filter,  $j$ 
   ( $j = 1, \dots, 4$ ) using (17).
4   for  $j = 1, \dots, 4$  do
5     - Perform the time update of the PDA filter for model  $j$  using (7).
6     - Perform the measurement validation using (8).
7     - Compute the association probabilities using (9).
8     if  $n(k) > 0$  then
9       - Perform the measurement update of the PDA filter for model  $j$  using
       (13).
10    else
11      - Operate the MVFF (26)–(36) and obtain  $\hat{\mathbf{x}}^*(k|k-1)$  and  $\mathbf{P}^*(k|k-1)$ .
12      - Reset the PDA filter for model  $j$  using (40).
13    end if
14  end for
15  - Compute the likelihoods using (21).
16  - Update the mode probabilities using (23).
17  - Obtain the ultimate estimate and covariance,  $\hat{\mathbf{x}}(k|k)$  and  $\mathbf{P}(k|k)$  using (25).
18 end

```

5. Simulation Results

In this section, the simulation results are presented to demonstrate the performance of the HIPFF. In the simulation, the 2D positions and velocities of a preceding vehicle were estimated using both the HIPFF and the IMMPDAF. The simulation scenario was as follows. Preceding and host vehicles were traveling at the same velocity on the highway. Subsequently, the preceding vehicle suddenly decelerated, and eventually stopped. The FCW system of the host vehicle estimated the position and velocity of the decelerating preceding vehicle. Various values of the initial velocities and relative distance of the vehicles were tested.

Radar clutters (false measurements) were generated independently, and were uniformly distributed in a square centered at the true measurement. The area of the square, \bar{V} , was computed using:

$$\begin{aligned} \bar{V} &= N_F / \lambda, \\ N_F &= [10\pi\gamma|S(k)|^{1/2}\lambda + 1]^{-}, \end{aligned} \quad (41)$$

where N_F is the number of false measurements and λ is the clutter density. In addition, the notation $[\cdot]^{-}$ corresponds to “rounded down to the nearest integer”, and $S(k)$ was obtained using a standard Kalman filter under the assumption of an ideal (no clutters) environment [17].

Automotive radars can detect objects within a field of view (FOV). In the simulation, it was assumed that a Delphi Electronically Scanning Radar (ESR) was used for the FCW system. The Delphi ESR is an automotive frequency modulated continuous wave (FMCW) radar. The maximum detection range and FOV of the Delphi ESR in the long-range mode are 174 m and 20° , respectively. In addition, the maximum detection range and FOV in the mid-range mode are 60 m and 90° , respectively.

The sampling interval, target detection probability, gate probability, and gate threshold were set as $T = 0.1$ s, $P_D = 0.9$, $P_G = 0.99$, and $\gamma = 9.21$, respectively. The noise variances of the range/bearing measurements were set as $\sigma_r = 0.25$ m and $\sigma_{theta} = 1.5^\circ$ [9,10]. Trajectories of preceding vehicle were generated using the DWNA model (1) with the standard deviation of process noise, $\sigma_w = 0.08$. For the four motion models, M_1 – M_4 , used for IMM estimation, the standard deviations were set to 1, 1, 10, and 0.1, respectively, and the horizon size of the MVFF was set as $N = 4$.

The performance of the estimation algorithms was evaluated using the root-mean-square position error (RMSPE) and the root-mean-square velocity error (RMSVE), which can be calculated using:

$$\text{RMSPE}(k) = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \sqrt{(x(k) - \hat{x}^i(k))^2 + (y(k) - \hat{y}^i(k))^2}, \quad (42)$$

$$\text{RMSVE}(k) = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \sqrt{(v_x(k) - \hat{v}_x^i(k))^2 + (v_y(k) - \hat{v}_y^i(k))^2}, \quad (43)$$

where N_{MC} is the number of Monte Carlo (MC) simulations; $(x(k), y(k))$ and $(v_x(k), v_y(k))$ are the true 2D positions and velocities, respectively; and $(\hat{x}^i(k), \hat{y}^i(k))$ and $(\hat{v}_x^i(k), \hat{v}_y^i(k))$ are the estimated 2D positions and velocities obtained in the i -th MC run, respectively. The RMSPE and RMSVE values were computed using 1000 MC runs. In addition, the percentage of lost track among the 1000 tracks was computed. A lost track was determined if the mode probability of M_1 exhibited the highest value for more than 20% of the simulation time.

Figures 2 and 3 show the simulation results of a successful track case when the initial distance between the preceding and host vehicles along the x-axis was $d_{0,x} = 100$ m, and the initial velocities along the x-axis was $v_{0,x} = 80$ km/h. The image revealed that the IMMPDAF did not diverge and successfully tracked the preceding vehicle under this condition. The preceding vehicle traveled at a velocity of 80 km/h for 2 s, after which it rapidly decelerated for 2 s and eventually stopped. The IMMPDAF did not always diverges and sometimes exhibited a successful track. However, under rapid deceleration, the IMMPDAF diverged often. If failures of IMMPDAF did not occur, the assisting MVFF did not operate. Subsequently, the HIPFF operated like a pure IMMPDAF and exhibited the same results as the IMMPDAF, as shown in Figure 2a–c. Figure 2d shows that a reset by the assisting MVFF did not occur. In addition, Figure 3a–d show the true/estimated positions and velocities as a function of time.

Figures 4 and 5 show the simulation results of a divergent track case, where the IMMPDAF lost the preceding vehicle and the RMSPE of the IMMPDAF diverged. In this simulation, the initial distance between the preceding and host vehicles was $d_{0,x} = 120$ m, and the initial velocity of the vehicles was $v_{0,x} = 100$ km/h. The preceding vehicle traveled at a velocity of 100 km/h for 2 s, and rapidly decelerated for 2.5 s and eventually stopped. In Figure 4, the HIPFF did not diverge and successfully tracked the preceding vehicle. The RMSPE and RMSVE values of the HIPFF increased initially, after which it decreased owing to the reset by the MVFF (Figure 4b,c). Figure 4d shows that the resets occurred three times. Figure 5a–d show the true/estimated positions and velocities as a function of time.

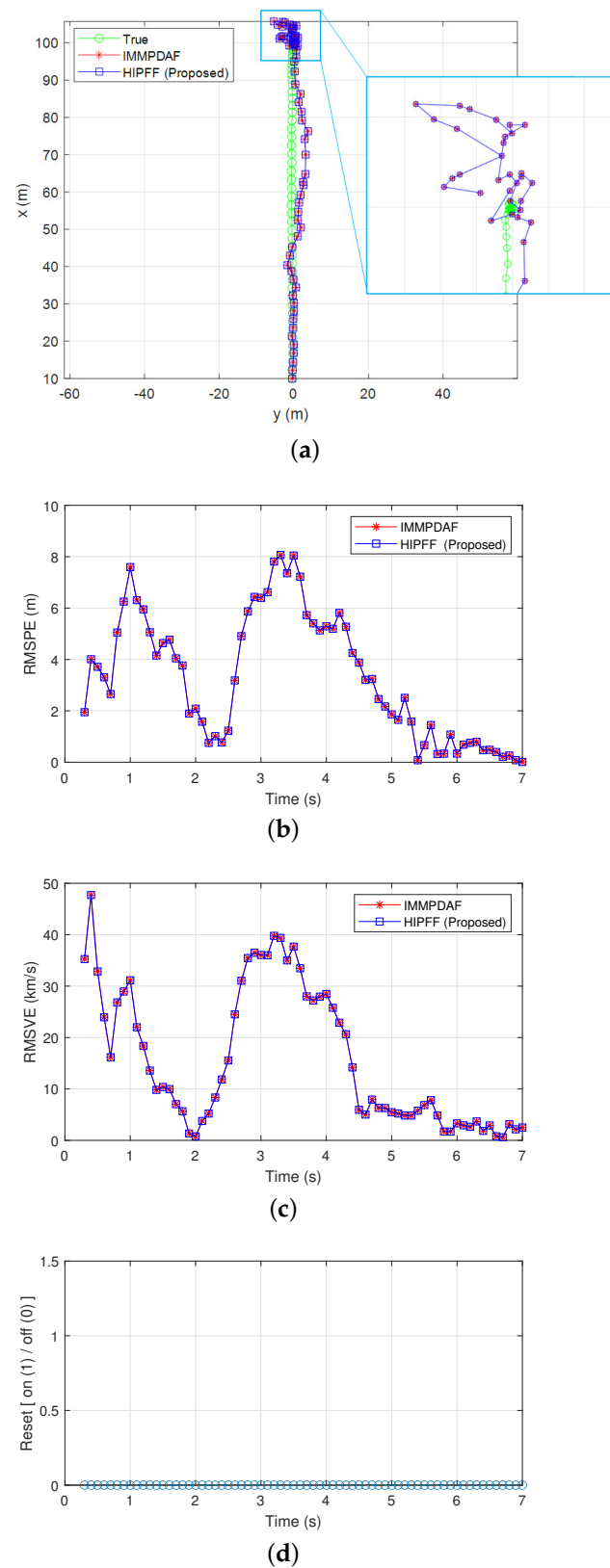
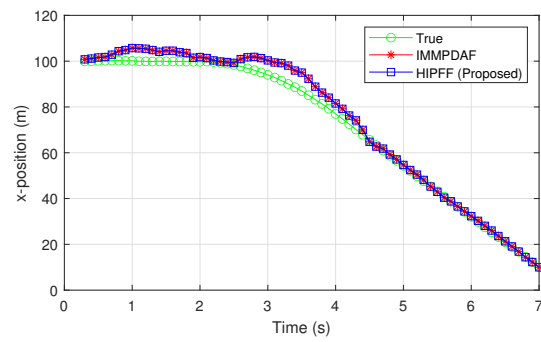
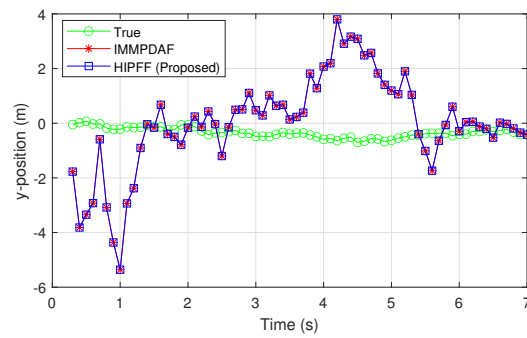


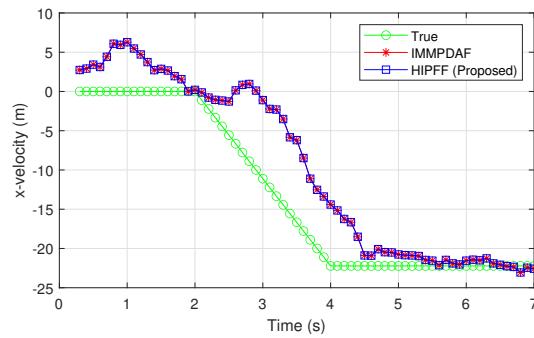
Figure 2. Simulation results of a successful track case: (a) true and estimated positions; (b) root-mean-square position error (RMSPE); (c) root-mean-square velocity error (RMSVE); and (d) reset timing.



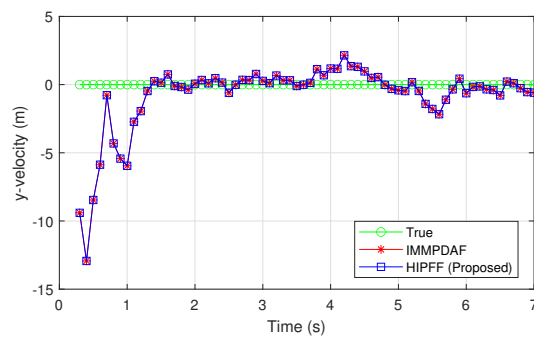
(a)



(b)

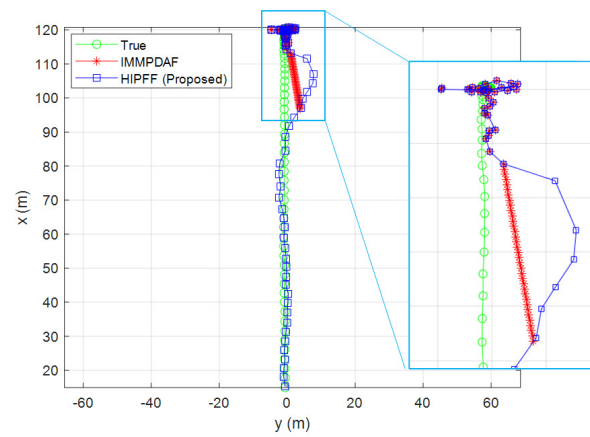


(c)

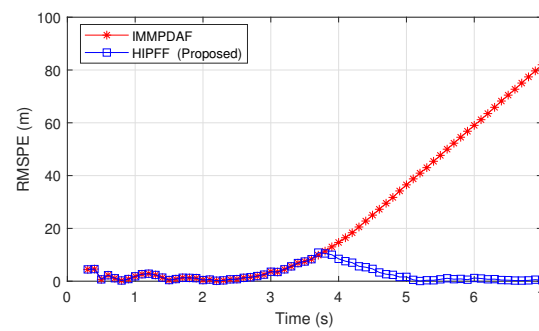


(d)

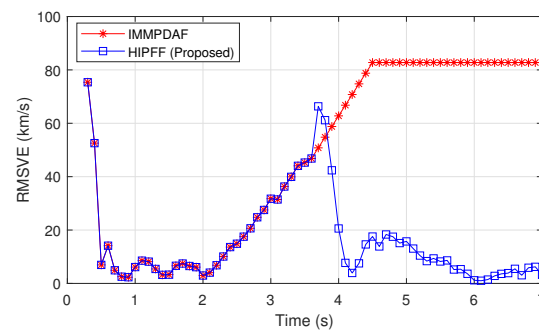
Figure 3. Estimation results of a successful track case: (a) x-position; (b) y-position; (c) x-velocity; and (d) y-velocity.



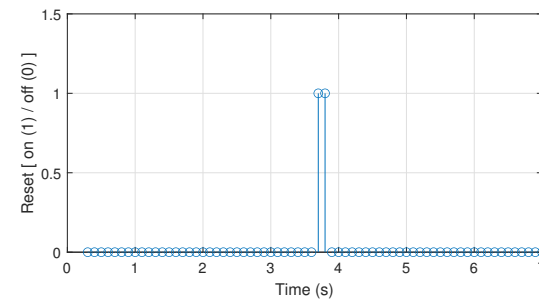
(a)



(b)



(c)



(d)

Figure 4. Simulation results in case of divergent interacting multiple model probabilistic data association filter (IMMPDAF): (a) true and estimated positions; (b) RMSPE; (c) RMSVE; and (d) reset timing.

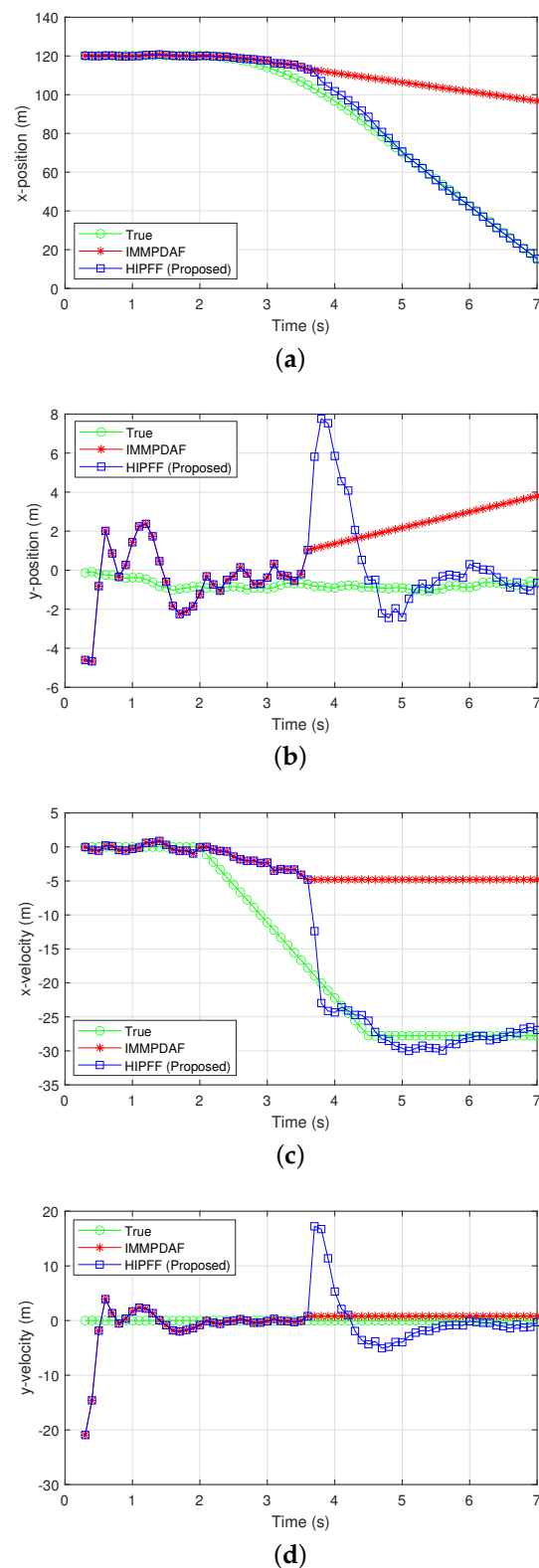


Figure 5. Estimation results of a successful track case: (a) x-position; (b) y-position; (c) x-velocity; and (d) y-velocity.

Tables 1–3 show the MC simulation results under two different clutter levels: heavy and medium. $\lambda = 5 \times 10^{-2}$ and $\lambda = 1 \times 10^{-2}$ were used for the heavy and medium clutter conditions, respectively. In addition, three different initial distance and velocities were tested. Tables 1–3 show the time-averaged RMSPE and RMSVE values, and the

percentages of the divergent tracks. The proposed HIPFF exhibited a higher accuracy than IMMPDAF in terms of the averaged RMSPE and RMSVE values. Moreover, the HIPFF was significantly more reliable than the IMMPDAF in terms of the percentage of divergent tracks. This demonstrates that the HIPFF is significantly more accurate and reliable than the conventional IMMPDAF under severe conditions.

Table 1. Monte Carlo (MC) simulation results under the condition of $v_{x,0} = 120$ km/h, $d_{x,0} = 140$ m.

Clutter Level	Heavy		Medium	
	IMMPDAF	HIPFF	IMMPDAF	HIPFF
Averaged RMSPE (m)	24.25	4.98	21.89	4.59
Averaged RMSVE (km/h)	53.98	33.94	49.39	47.23
Percentage of divergent track	95.4	0	91.7	2.7

Table 2. MC simulation results under condition of $v_{x,0} = 100$ km/h, $d_{x,0} = 120$ m.

Clutter Level	Heavy		Medium	
	IMMPDAF	HIPFF	IMMPDAF	HIPFF
Averaged RMSPE (m)	22.21	4.29	19.29	3.7
Averaged RMSVE (km/h)	47.06	30.93	41.68	38.1
Percentage of divergent track	92.4	0	87.6	0.6

Table 3. MC simulation results under condition of $v_{x,0} = 80$ km/h, $d_{x,0} = 100$ m.

Clutter Level	Heavy		Medium	
	IMMPDAF	HIPFF	IMMPDAF	HIPFF
Averaged RMSPE (m)	19.39	3.63	14.76	2.74
Averaged RMSVE (km/h)	39.15	28.2	31.37	29.49
Percentage of divergent track	88.9	0	75	1.7

6. Conclusions

This study proposed a novel state estimation algorithm, namely hybrid IMM-PDA/FIR filter (HIPFF), for forward collision warning (FCW) systems based on automotive radars. The proposed algorithm utilized interacting multiple model probabilistic data association filter (IMMPDAF) as the main filter, and a finite impulse response (FIR) filter (i.e., minimum variance FIR filter (MVFF)) for recovering the main filter from failures. Under a rapidly decelerating preceding vehicle condition, the simulation results revealed that HIPFF exhibited significantly accurate and reliable estimation of position/velocity, compared to that of the pure IMMPDAF. Therefore, the HIPFF algorithm is expected to exhibit improved performance for automotive radar-based FCW systems.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2020R1F1A1072428).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

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

Abbreviations

The following abbreviations are used in this manuscript:

ADAS	Advanced Driver-Assistance Systems
FCW	Forward Collision Warning
GPS	Global Positioning System
TTC	Time-To-Collision
FIR	Finite Impulse Response
PDA	Probabilistic Data Association
IMM	Interacting Multiple Model
IMMPDAF	Interacting Multiple Model Probabilistic Data Association Filter
HIPFF	Hybrid IMM-PDA/FIR Filter
2D	Two-Dimensional
KF	Kalman Filter
DWNA	Discrete White Noise Acceleration
MVFF	Minimum Variance FIR Filter
FOV	Field of View
ESR	Electronically Scanning Radar
RMSPE	Root-Mean-Square Position Error
RMSVE	Root-Mean-Square Velocity Error

References

1. Khan, M.Q.; Lee, S. A Comprehensive Survey of Driving Monitoring and Assistance Systems. *Sensors* **2019**, *19*, 2574. [\[CrossRef\]](#) [\[PubMed\]](#)
2. Baek, M.; Jeong, D.; Choi, D.; Lee, S. Vehicle Trajectory Prediction and Collision Warning via Fusion of Multisensors and Wireless Vehicular Communications. *Sensors* **2020**, *20*, 288. [\[CrossRef\]](#) [\[PubMed\]](#)
3. Gonzalez-Partida, J.T.; Leon-Infante, F.; Blazquez-Garcia, R.; Burgos-Garcia, M. On the Use of Low-Cost Radar Networks for Collision Warning Systems Aboard Dumpers. *Sensors* **2014**, *14*, 3921–3938. [\[CrossRef\]](#) [\[PubMed\]](#)
4. Song, W.; Yang, Y.; Fu, M.; Li, Y.; Wang, M. Lane Detection and Classification for Forward Collision Warning System Based on Stereo Vision. *IEEE Sens. J.* **2018**, *18*, 5151–5163. [\[CrossRef\]](#)
5. Barrios, C.; Motai, Y. Improving estimation of vehicle's trajectory using the latest global positioning system with Kalman filtering. *IEEE Trans. Instrum. Meas.* **2011**, *60*, 3747–3755. [\[CrossRef\]](#)
6. Gehrig, S.K.; Stein, F.J. Collision Avoidance for Vehicle-Following Systems. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 233–244. [\[CrossRef\]](#)
7. Vahidi, A.; Eskandarian, A. Research Advanced in Intelligent Collision Avoidance and Adaptive Cruise Control. *IEEE Trans. Intell. Transp. Syst.* **2003**, *4*, 143–153. [\[CrossRef\]](#)
8. Fleming, B. Recent Advancement in Automotive Radar Systems Automotive Electronics. *IEEE Veh. Technol. Mag.* **2012**, *7*, 4–9. [\[CrossRef\]](#)
9. Han, S.K.; Ra, W.S.; Whang, I.H.; Park, J.B. Gaussian Mixture Approach to Decision Making for Automotive Collision Warning Systems. *Int. J. Control Autom. Syst.* **2015**, *13*, 1182–1192. [\[CrossRef\]](#)
10. Han, S.K.; Ra, W.S.; Whang, I.H.; Park, J.B. Linear Recursive Automotive Target Tracking Filter for Advanced Collision Warning Systems. *Appl. Math. Inf. Sci.* **2014**, *8*, 1145–1151. [\[CrossRef\]](#)
11. Tu, P.J.; Kiang, J.F. Estimation on location, velocity, and acceleration with high precision for collision avoidance. *IEEE Trans. Intell. Transp. Syst.* **2010**, *8*, 157–167.
12. Lee, M.S. An efficient multitarget tracking algorithm for car applications. *IEEE Trans. Ind. Electron.* **2003**, *50*, 397–399.
13. Yang, W.; Wan, B.; Qu, X. A Forward Collision Warning System Using Driving Intention Recognition of the Front Vehicle and V2V Communication. *IEEE Access* **2020**, *8*, 11268–11278. [\[CrossRef\]](#)
14. Iranmanesh, S.M.; Mahjoub, H.N.; Kazemi, H.; Fallah, Y.P. An Adaptive Forward Collision Warning Framework Design Based on Driver Distraction. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 3925–3934. [\[CrossRef\]](#)
15. Wang, J.; Yu, C.; Li, S.E.; Wang, L. A Forward Collision Warning Algorithm with Adaptation to Driver Behaviors. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 1157–1167. [\[CrossRef\]](#)
16. Bar-Shalom, Y.; Tse, E. Tracking in a Cluttered Environment with Probabilistic Data Association. *Automatica* **1975**, *11*, 451–460. [\[CrossRef\]](#)
17. Bar-Shalom, Y.; Willett, P.K.; Tian, X. *Tracking and Data Fusion—A Handbook of Algorithms*; YBS Publishing: Storrs, CT, USA, 2011.
18. Kodagoda, K.R.S.; Ge, S.S.; Wijesoma, W.S.; Balasuriya, A.P. IMMPDAF Approach for Road-Boundary Tracking. *IEEE Trans. Veh. Technol.* **2007**, *56*, 478–486. [\[CrossRef\]](#)
19. Kirubarajan, T.; Bar-Shalom, Y.; Blair, W.D.; Watson, G.A. IMMPDAF for radar management and tracking benchmark with ECM. *IEEE Trans. Aerosp. Electron. Syst.* **1998**, *34*, 1115–1134. [\[CrossRef\]](#)

20. Raghavan, V.; Pattipati, K.R.; Bar-Shalom, Y. Efficient L-D factorization algorithms for PDA, IMM, and IMMPDA filters. *IEEE Trans. Aerosp. Electron. Syst.* **1993**, *29*, 1297–1310. [[CrossRef](#)]
21. Forthmann, T.; Bar-Shalom, Y.; Scheffe, M. Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association. *IEEE J. Ocean. Eng.* **1983**, *8*, 173–184. [[CrossRef](#)]
22. Blackman, S.S. Multiple Hypothesis Tracking for Multiple Target Tracking. *IEEE Trans. Aerosp. Electron. Syst.* **2004**, *19*, 5–18. [[CrossRef](#)]
23. Mahler, R.P.S. *Statistical Multisource-Multitarget Information Fusion*; Artech House: Norwood, MA, USA, 2007.
24. You, S.H.; Ahn, C.K.; Zhao, S.; Shmaliy, Y.S. Frobenius Norm-Based Unbiased Finite Impulse Response Fusion Filtering for Wireless Sensor Networks. *IEEE Trans. Ind. Electron.* **2022**, *69*, 1867–1876. [[CrossRef](#)]
25. Zhao, S.; Shmaliy, Y.S.; Ahn, C.K.; Luo, L. An Improved Iterative FIR State Estimator and Its Applications. *IEEE Trans. Ind. Inform.* **2020**, *16*, 1003–1012. [[CrossRef](#)]
26. Xu, Y.; Ahn, C.K.; Shmaliy, Y.S.; Chen, X.; Bu, L. Indoor INS/UWB-based human localization with missing data utilizing predictive UFIR filtering. *IEEE/CAA J. Autom. Sin.* **2019**, *6*, 952–960. [[CrossRef](#)]
27. Pak, J.M.; Ahn, C.K. Dual-Mode Deadbeat H_2 FIR Filtering for Discrete-Time Systems. *Signal Process.* **2021**, *184*, 108057. [[CrossRef](#)]
28. Pak, J.M.; Ahn, C.K.; Shi, P.; Shmaliy, Y.S.; Lim, M.T. Distributed hybrid particle/FIR filtering for mitigating NLOS effects in TOA-based localization using wireless sensor networks. *IEEE Trans. Ind. Electron.* **2017**, *64*, 5182–5191. [[CrossRef](#)]
29. Pak, J.M.; Ahn, C.K.; Shmaliy, Y.S.; Lim, M.T. Improving reliability of particle filter-based localization in wireless sensor networks via hybrid particle/FIR filtering. *IEEE Trans. Ind. Inform.* **2015**, *11*, 1089–1098. [[CrossRef](#)]
30. Pak, J.M.; Ahn, C.K.; Shmaliy, Y.S.; Shi, P.; Lim, M.T. Switching extensible FIR filter bank for adaptive horizon state estimation with application. *IEEE Trans. Control Syst. Technol.* **2016**, *24*, 1052–1058. [[CrossRef](#)]
31. Pak, J.M.; Ahn, C.K.; Lim, M.T.; Song, M.K. Horizon group shift FIR filter: Alternative nonlinear filter using finite recent measurements. *Measurement* **2014**, *57*, 33–45. [[CrossRef](#)]