IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 1, 2021, accepted August 18, 2021, date of publication August 24, 2021, date of current version September 1, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3107464

Hybrid PDA/FIR Filtering for Preceding Vehicle Tracking Using Automotive Radars

JUNG MIN PAK, (Member, IEEE)
Department of Electrical Engineering, Wonkwang University, Iksan-si, Jeollabuk-do 54538, Republic of Korea

e-mail: destin11@wku.ac.kr

This work was supported by the National Research Foundation of Korea (NRF) through Korea Government (MSIT) under Grant 2020R1F1A1072428.

ABSTRACT This paper proposes a novel single vehicle tracking algorithm with enhanced reliability for automotive radar systems. The proposed algorithm overcomes the weaknesses of the probabilistic data association filter (PDAF) in single-target tracking in clutter. The PDAF is successful in normal situations, but may fail to track a target owing to various factors, such as the initialization errors and the sudden changes in the target motion. The proposed algorithm can recover the PDAF from failures using an assisting finite impulse response (FIR) filter. The FIR filter operates only when the PDAF cannot track a target properly, and additionally offers state estimate and estimation error covariance to reset the PDAF. The proposed algorithm, the hybrid PDAF/FIR filter (HPFF), combines the PDAF and FIR filter, and hence shows enhanced reliability. Simulations of preceding vehicle tracking using an automotive radar demonstrate the effect and performance of the proposed HPFF.

INDEX TERMS Automotive radar, finite impulse response filter, FIR filter, hybrid PDAF/FIR filter, probabilistic data association filter.

I. INTRODUCTION

Intelligent cars equipped with advanced driver assistance systems (ADAS) have become common in recent years, and automotive radars have become increasingly important as an essential sensor for ADAS. Automotive radars are mainly used in adaptive cruise control and forward collision warning systems for tracking forward vehicles and pedestrians [1]. Because radar measurements are affected by noise and false measurements (clutter), filtering and data association algorithms are used to improve the tracking accuracy [2]–[4]. The most popular algorithm is the probabilistic data association filter (PDAF) [5], which has been successful in diverse radar tracking applications over the past few decades [6], [7].

Tracking a target in the presence of false measurements requires determining which measurements are correct. The nearest neighbor standard filter (NNSF) solves this problem by selecting the nearest measurement according to a certain distance measure. On the other hand, the PDAF is based on the probabilistic data association (PDA) algorithm, which calculates the association probabilities for each measurement and combines the measurements based on the probabilities. As compared to the NNSF, the PDAF provides successful tracking performance, particularly in dense clutter environments [6]. However, the PDAF algorithm sometimes fails because of various factors, such as large initialization errors and sudden changes in the target motion. If such failures occur, the PDAF cannot recover itself and diverges, leading to very large tracking errors.

Recently, finite impulse response (FIR) filters [8]–[17] have gained attention from researchers because of their relative benefits. Bayes filters, including the PDAF, are recursive filters, meaning that they require an initialization process and use all past measurements to produce the current state estimate. On the other hand, FIR filters are batch processing filters that do not require initialization and use only the recent finite measurements. Owing to their unique filter structure, FIR filters exhibit fast response with respect to abrupt changes in the motion of the target and are robust against filter divergence. Various FIR filters have been developed and successfully applied to tracking problems, such as localization, visual object tracking, and target tracking. Especially, the hybrid particle/FIR filter [11], [12] is noteworthy because it overcomes the main drawback of a particle

The associate editor coordinating the review of this manuscript and approving it for publication was Shadi Alawneh.

filter (PF) by employing an assisting FIR filter. In nonlinear state estimation problems, the PF is superior to other filters, but can fail and diverge owing to the sample impoverishment phenomenon. This occurs under certain conditions, for example, low process/measurement noise and a small number of particles. The hybrid particle/FIR filter algorithm detects PF failures and operates the assisting FIR filter. The failed PF is reset and rebooted using the information obtained from the assisting FIR filter.

Herein, we propose a novel hybrid filtering algorithm, called the hybrid PDA/FIR filter (HPFF), to overcome the limitations of the PDAF. The PDAF acts as the main filter because it is superior to other filters in single target tracking in clutter. The failure of the PDAF usually occurs when there is no measurement within the validation region. Because there is no available measurement, the PDAF cannot perform the measurement update process, which leads to tracking failure. In such cases, the HPFF operates the assisting FIR filter and obtains the state estimate and estimation error covariance. With this information, the main filter, PDAF, is reset and rebooted. The proposed HPFF is applied to tracking a preceding vehicle using an automotive radar in a clutter environment. Through simulation, the ability of the HPFF to recover from failure is demonstrated by comparing its performance with that of the conventional PDAF.

The main contributions of this paper are as follows. There have been various algorithms that improve the performance of the PDAF. However, to the best of our knowledge, there is no method to recover a failed PDAF. The proposed HPFF is the first algorithm to recover the PDAF from such failures. In addition, a new FIR filter for target tracking in clutter has been proposed for the first time. The HPFF was successfully applied to vehicle tracking using an automotive radar. Automotive radars should maintain reliability in clutter environments, and the HPFF can provide reliable vehicle tracking.

The rest of this paper is organized as follows. In Section II, PDAF-based preceding vehicle tracking using an automotive radar is explained. In Section III, the proposed HPFF algorithm is discussed. In Section IV, the simulation results obtained under harsh conditions are discussed to demonstrate the capability of the HPFF. Finally, conclusions are drawn in Section V. All the acronyms used in this paper are defined in Table 1.

TABLE 1. List of acronyms.

Acronym	Explanation
ADAS	Advanced Driver Assistance System
FIR	Finite Impulse Response
HPFF	Hybrid PDA/FIR Filter
IIR	Infinite Impulse Response
MC	Monte Carlo
MVFF	Minimum Variance FIR Filter
NNSF	Nearest Neighbor Standard Filter
PDA	Probabilistic Data Association
PDAF	Probabilistic Data Association Filter
PF	Particle Filter
RMS	Root Mean Square
RTAMS	Root Time Averaged Mean Square

II. PDAF-BASED PRECEDING VEHICLE TRACKING USING AUTOMOTIVE RADAR

In this paper, an automotive radar system for tracking a preceding vehicle is considered. The automotive radar is mounted in the middle front of the host vehicle and measures the relative range and bearing angle of a preceding vehicle. The position of a preceding vehicle is represented in a two-dimensional (2D) Cartesian coordinate system, as shown in Fig. 1. The origin of the coordinate system is the location of the automotive radar and the x-axis is aligned with the longitudinal direction of the host vehicle. The 2D position of a preceding vehicle in the Cartesian coordinate system is represented as (x, y), and hence, the relative range from the radar is

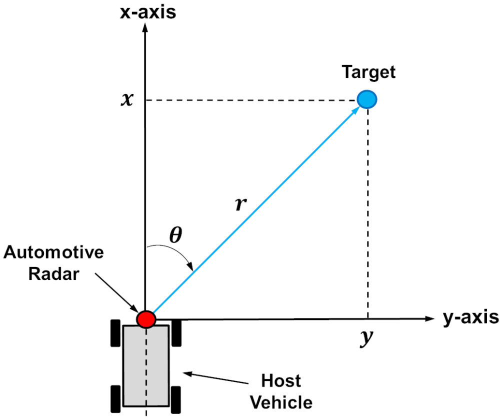


FIGURE 1. Automotive radar geometry.

$$r = \sqrt{x^2 + y^2}. \tag{1}$$

The bearing is measured clockwise from the positive x-axis and can be represented as

$$\theta = tan^{-1}(\frac{y}{x}). \tag{2}$$

The automotive radar provides the measured range and bearing, r_m and θ_m, respectively, which are defined with respect to the true range and bearing, r and θ as

$$\begin{aligned} r_m &= r + v_r, \\ \theta_m &= \theta + v_\theta, \end{aligned} \tag{3}$$

where v_r and v_θ are the measurement noises, which are assumed to be independent and zero mean white Gaussian noises with covariances σ_r^2 and σ_θ^2, respectively [6].

The measured range and bearing can be converted to Cartesian coordinate as follows:

$$\begin{aligned} x_m &= r_m\, cos(\theta_m), \\ y_m &= r_m\, sin(\theta_m). \end{aligned} \tag{4}$$

The noise covariances of the Cartesian coordinate measurements are approximately computed as

$$\begin{aligned}\mathbf{R} &= \begin{bmatrix} R^{11} & R^{12} \\ R^{12} & R^{22} \end{bmatrix},\\ R^{11} &= r_m^2\, \sigma_\theta^2\, sin^2(\theta_m) + \sigma_r^2\, cos^2(\theta_m),\\ R^{22} &= r_m^2\, \sigma_\theta^2\, cos^2(\theta_m) + \sigma_r^2\, sin^2(\theta_m),\\ R^{12} &= (\sigma_r^2 - r_m^2\, \sigma_\theta^2)\, sin(\theta_m)\, cos(\theta_m). \end{aligned} \tag{5}$$

Note that the above conversion from the polar coordinates to the Cartesian coordinates is valid when the following condition is satisfied [6]:

$$\frac{r\, \sigma_\theta^2}{\sigma_r^2} < 0.4, \tag{6}$$

where σ_r^2 is in radians.

The target position in the Cartesian coordinates can be obtained from radar measurements via polar to Cartesian coordinate conversion, but it is not sufficiently accurate because of conversion (linearization) errors and measurement noise. Thus, a stochastic filter, also referred to as a state estimator, is used to improve the accuracy in obtaining a target position. To use a filter, a motion model is required. The constant velocity (CV) motion model is one of the most popular models used in radar tracking. The CV motion model assumes that the target velocity is constant within the sampling interval and the acceleration is a zero-mean white Gaussian random variable. The 2D positions and velocities in the Cartesian coordinates at time step k are combined by defining a state vector as $\mathbf{x}_k = [x_k\ y_k\ \dot{x}_k\ \dot{y}_k]^T$. Then, the CV model is represented as follows [18], [19]:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{G}\mathbf{w}_k, \tag{7}$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{bmatrix}, \tag{8}$$

where T is the sampling interval and $\mathbf{w}_k$ is the white Gaussian noise that represents the acceleration. The covariance of $\mathbf{w}_k$ is $\mathbf{Q}_k = \sigma_w^2 \mathbf{I}_2$, where $\mathbf{I}_2$ is the 2×2 identity matrix. In the CV model, $\mathbf{Q}_k$ is a key parameter that determines the characteristics of the motion of the target. Thus, $\mathbf{Q}_k$ should be carefully designed, and if not, the accuracy of the filtered position may be degraded. A practical design of $\mathbf{Q}_k$ satisfies $0.5\, a_M \leq \sigma_w \leq a_M$, where a_M is the maximum acceleration magnitude of a target.

The filtering process requires another model, the measurement model. Two types of measurements, (r_m, θ_m) and (x_m, y_m), are available in radar tracking. Using the raw measurements (r_m, θ_m), the measurement model that describes the relation between the measurements and the state vector becomes nonlinear. In this case, nonlinear filters, such as the extended Kalman filter and PF, should be used. Using the converted measurements (x_m, y_m), the measurement model becomes linear, and linear filters, such as the Kalman filter, can be used. It is known that linear filtering using the converted measurements is generally more accurate than nonlinear filtering using the raw measurements [6]. Thus, a linear measurement model using (x_m, y_m) is used in this study, and is represented as follows [6]:

$$\mathbf{z}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k, \tag{9}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \tag{10}$$

where $\mathbf{z}_k$ is the measurement vector defined as $\mathbf{z}_k = [x_m\, y_m]^T$. The measurement noise $\mathbf{v}_k$ is assumed to be zero-mean white Gaussian, and its covariance is computed using (5).

With the converted measurements and the motion/measurement models, the positions of a preceding vehicle can be estimated via linear filtering. However, radar measurements include not only correct measurements that come from the target, but also false measurements (clutter) that come from the road, guardrail, the radar echo of other vehicles, etc. The PDAF [5] solves this problem via association between the measurements and the estimated position. The PDAF algorithm performs time updates, measurement validation, data association, and measurement updates at each time step. The time update process predicts the state at the current time k using the motion model as follows [5], [6]:

$$\begin{aligned}\hat{\mathbf{x}}_k^- &= \mathbf{A}\hat{\mathbf{x}}_{k-1}^+,\\ \mathbf{P}_k^- &= \mathbf{A}\mathbf{P}_{k-1}^+\mathbf{A}^T + \mathbf{Q}_k, \end{aligned} \tag{11}$$

where $\mathbf{x}_k^-$ and $\mathbf{P}_k^-$ are the *a priori* (i.e., the measurements are not used yet) estimated state and estimation error covariance.

The measurement validation process selects the measurements within the validation region (also called the gate) as follows [5], [6], [20], [21]:

$$\begin{aligned}\nu_k(i) = (\mathbf{z}_k(i) - \hat{\mathbf{z}}_k)^T \mathbf{S}_k^{-1}(i)\\ \times (\mathbf{z}_k(i) - \hat{\mathbf{z}}_k) \leq \gamma, \quad i = 1, \ldots, m(k), \end{aligned} \tag{12}$$

where $\mathbf{z}_k(i)$ is the i-th measurement; $m(k)$ is the number of measurements; $\nu_k(i)$ is the innovation for $\mathbf{z}_k(i)$; γ is the gate threshold; $\hat{\mathbf{z}}_k = \mathbf{C}\hat{\mathbf{x}}_k^-$ is the predicted measurement; and $\mathbf{S}_k(i)$ is the innovation covariance, which is defined as

$$\mathbf{S}_k(i) = \mathbf{C}\mathbf{P}_k^-\mathbf{C}^T + \mathbf{R}_k(i), \tag{13}$$

where $\mathbf{R}_k(i)$ can be obtained using $\mathbf{z}_k(i)$ via the measurement conversion using (4) and (5). The gate threshold γ can be obtained from a *chi-square* table. For example, γ is 9.21 when the gate probability P_G is 0.99 and the dimension of the measurement vector n_z is 2.

The data association process, called the PDA, provides the association probability for the measurements within the validation region. If the number of measurements in the validation region at time k is $n(k)$, the association probability for

$\mathbf{z}_k(i)$ being the correct measurement is computed as

$$\beta_k(i) = \begin{cases} \dfrac{\mathcal{L}_k(i)}{1 - P_D P_G + \sum_{j=1}^{n(k)} \mathcal{L}_k(j)} & i = 1, \ldots, n(k) \\ \dfrac{1 - P_D P_G}{1 - P_D P_G + \sum_{j=1}^{n(k)} \mathcal{L}_k(j)} & i = 0 \end{cases} \quad (14)$$

where $i = 0$ means "none is correct", and $n(k)$ is the number of validated measurements [5], [6], [21]. The likelihood ratio $\mathcal{L}_k(j)$ is computed as

$$\mathcal{L}_k(j) = \frac{exp[-0.5(\mathbf{z}_k(j) - \hat{\mathbf{z}}_k)^T \mathbf{S}_k^{-1}(j)(\mathbf{z}_k(j) - \hat{\mathbf{z}}_k)]}{(2\pi)^{n_z/2}|\mathbf{R}_k(j)|^{1/2}\, n(k)} \times P_D\, V_k(j), \quad (15)$$

where P_D is the detection probability, and $V_k(j)$ is the volume of the validation region, which is computed as

$$V_k(j) = c_{n_z} \gamma^{n_z/2} |\mathbf{S}_k(j)|^{1/2}. \quad (16)$$

The measurement update process corrects the *a priori* estimate using the measurements as

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \sum_{i=1}^{n(k)} \mathbf{W}_k(i)\, \beta_k(i)\, \nu_k(i), \quad (17)$$

where $\mathbf{W}_k(i)$ and $\nu_k(i)$ are the filter gain and combined innovation, respectively, which are defined as follows [21]:

$$\mathbf{W}_k(i) = \mathbf{P}_k^- \mathbf{C}^T \mathbf{S}_k^{-1}(i), \quad (18)$$
$$\nu_k(i) = \mathbf{z}_k(i) - \hat{\mathbf{z}}_k. \quad (19)$$

The estimation error covariance is also updated as follows [21]:

$$\mathbf{P}_k^+ = \bar{\mathbf{P}}_k + \tilde{\mathbf{P}}_k, \quad (20)$$
$$\bar{\mathbf{P}}_k = \beta_k(0)\, \mathbf{P}_k^- + \sum_{i=1}^{n(k)} \beta_k(i)(\mathbf{I} - \mathbf{W}_k(i)\mathbf{C})\mathbf{P}_k^-, \quad (21)$$
$$\tilde{\mathbf{P}}_k = \sum_{i=1}^{n(k)} \beta_k(i)\mathbf{W}_k(i)\nu_k(i)\nu_k^T(i)\mathbf{W}_k^T(i) - \left(\sum_{i=1}^{n(k)} \beta_k(i)\mathbf{W}_k(i)\nu_k(i) \right) \left(\sum_{i=1}^{n(k)} \beta_k(i)\nu_k^T(i)\mathbf{W}_k^T(i) \right). \quad (22)$$

The measurement update process cannot be performed if there is no measurement in the validation region. In that case, the PDAF performs only the time update process [5], [6].

III. HYBRID PDA/FIR FILTER

The PDAF cannot perform the measurement update when there is no measurement in the validation region. Fig. 2 shows a graphical illustration of that situation. The center of the validation region is the predicted measurement $\hat{\mathbf{z}}_k$ that is obtained from the *a priori* estimated state $\hat{\mathbf{x}}_k^-$. Clutters are distributed around the correct measurement. The target suddenly changes motion, and its actual location is far from the predicted position (i.e., $\hat{\mathbf{x}}_k^-$). In this case, measurements including clutters might be located out of the validation region centered on $\hat{\mathbf{z}}_k$. Hence, no measurement is in the validation region. The PDAF does not perform the measurement update, and the estimated position (state) becomes increasingly distant from the actual state. This phenomenon leads to the tracking failure (target loss) of the PDAF. In this section, the HPFF algorithm proposed to overcome the PDAF failure is discussed.

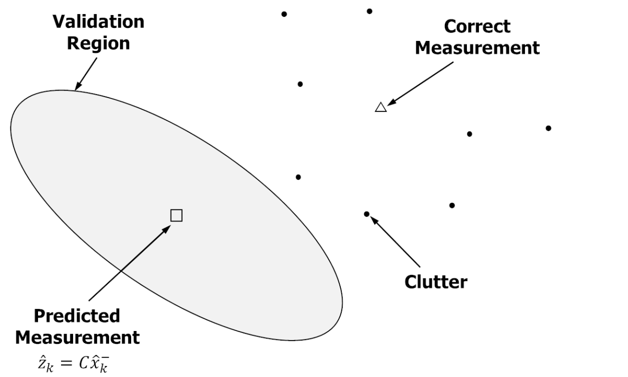


FIGURE 2. Situation in which no measurement is in the validation region.

The HPFF is a hybrid filter that combines a main filter (PDAF) and an assisting filter (FIR filter). The HPFF algorithm performs FIR filtering when there is no measurement in the validation region. Among the various FIR filters available, we adopt the minimum variance FIR filter (MVFF), because it can provide not only an estimated state, but also estimation error covariance, which are necessary to reset the main filter. In addition, the MVFF is a one-step prediction filter that produces $\hat{\mathbf{x}}_k$ using the past measurements on the time horizon $[k-N, k-1]$, where N is the horizon size. Thus, the MVFF can provide $\hat{\mathbf{x}}_k$ and $\mathbf{P}_k$ without using a current measurement that is less reliable in the situation of "no measurements in the validation region." To use the MVFF in the presence of clutters, the algorithm is modified as follows.

On the time horizon $[k-N, k-1]$, the most reliable measurements at each time steps are selected using the nearest neighbor algorithm.

$$\bar{\mathbf{z}}_h = \arg\min_{\mathbf{z}_h(i)} (\mathbf{z}_h(i) - \hat{\mathbf{z}}_h)^T \mathbf{S}_k^{-1}(i) \times (\mathbf{z}_h(i) - \hat{\mathbf{z}}_h), \quad i = 1, \ldots, m(h), \quad (23)$$
$$\hat{\mathbf{z}}_h = \mathbf{C}\hat{\mathbf{x}}_h^+, \quad (24)$$
$$\mathbf{S}_h = \mathbf{C}\mathbf{P}_h^+\mathbf{C}^T + \mathbf{R}_h, \quad (25)$$

where $h = k-N, k-N+1, \ldots, k-1$; $\mathbf{z}_h(i)$ is the i-th measurement at time h; and $m(h)$ is the number of measurements at time h. The measurement noise covariance $\mathbf{R}_h$ is obtained using $\bar{\mathbf{z}}_h$ via (4) and (5). $\hat{\mathbf{x}}_h^+$ and $\mathbf{P}_h^+$ are the estimated state and the estimation error covariance obtained by the PDAF, respectively.

Remark 1: The nearest neighbor measurement $\bar{\mathbf{z}}_h$ is obtained using the *a priori* estimation results, $\mathbf{x}_k^+$ and $\mathbf{P}_k^+$.

On the other hand, the measurement validation in both the PDAF and NNSF is performed using the *a posteriori* estimation results, $\hat{\mathbf{x}}_k^-$ and $\mathbf{P}_k^-$. Thus, $\bar{\mathbf{z}}_h$ is more reliable than the validated measurements used in the PDAF and NNSF. The assisting MVFF uses these reliable past measurements instead of the less reliable current measurement.

Using $\bar{\mathbf{z}}_h$ and $\mathbf{R}_h$ on the horizon $[k-N, k-1]$, the MVFF produces the state estimate $\hat{\mathbf{x}}_k^*$ and the estimation error covariance $\mathbf{P}_k^*$ as follows [12], [14]:

$$
\begin{aligned}
\hat{\mathbf{x}}_k^* &= \mathbf{L}\,\mathbf{Z}_N,\\
\mathbf{P}_k^* &= \mathbf{K}_N\mathbf{Q}_N\mathbf{K}_N^T + \mathbf{L}\mathbf{R}_N\mathbf{L}^T,\\
\mathbf{L} &= \mathbf{J}_N\begin{bmatrix}\mathbf{M}_{1,1} & \mathbf{M}_{1,2}\\ \mathbf{M}_{1,2}^T & \mathbf{M}_{2,2}\end{bmatrix}^{-1}\begin{bmatrix}\bar{\mathbf{C}}_N^T\\ \bar{\mathbf{G}}_N^T\end{bmatrix}\mathbf{R}_N^{-1},\\
\mathbf{J}_N &= [\mathbf{A}^N \;\; \mathbf{A}^{N-1} \;\; \mathbf{A}^{N-2} \;\; \ldots \;\; \mathbf{A} \;\; \mathbf{I}],\\
\mathbf{M}_{1,1} &= \bar{\mathbf{C}}_N^T\mathbf{R}_N^{-1}\bar{\mathbf{C}}_N,\\
\mathbf{M}_{1,2} &= \bar{\mathbf{C}}_N^T\mathbf{R}_N^{-1}\bar{\mathbf{G}}_N,\\
\mathbf{M}_{2,2} &= \bar{\mathbf{G}}_N^T\mathbf{R}_N^{-1}\bar{\mathbf{G}}_N + \mathbf{Q}_N^{-1},\\
\bar{\mathbf{C}}_N &= \begin{bmatrix}\mathbf{C}\\ \mathbf{CA}\\ \mathbf{CA}^2\\ \vdots\\ \mathbf{CA}^{N-1}\end{bmatrix},\\
\bar{\mathbf{G}}_N &= \begin{bmatrix}\mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0}\\ \mathbf{CG} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0}\\ \mathbf{CAG} & \mathbf{CG} & \ldots & \mathbf{0} & \mathbf{0}\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ \mathbf{CA}^{n-1}\mathbf{G} & \mathbf{CA}^{n-2}\mathbf{G} & \ldots & \mathbf{CG} & \mathbf{0}\end{bmatrix},\\
\mathbf{R}_N &= [\mathit{diag}(\overbrace{\mathbf{R}_m\;\mathbf{R}_{m+1}\;\ldots\;\mathbf{R}_n}^{N})],\\
\mathbf{Q}_N &= [\mathit{diag}(\overbrace{\mathbf{Q}_m\;\mathbf{Q}_{m+1}\;\ldots\;\mathbf{Q}_n}^{N})],\\
\mathbf{Z}_N &= [\mathbf{z}_m^T \;\; \mathbf{z}_{m+1}^T \;\; \cdots \;\; \mathbf{z}_n^T]^T,\\
\mathbf{K}_N &= [\mathbf{A}^{N-1}\mathbf{G} \;\; \mathbf{A}^{N-2}\mathbf{G} \;\; \ldots \;\; \mathbf{AG} \;\; \mathbf{G}],
\end{aligned}
\tag{26}
$$

where m and n are the initial and final points of the time horizon and are defined as $m = k-N$ and $n = k-1$, respectively.

Remark 2: The assisting MVFF (26) is an extended version of the MVFF for linear time-varying state-space models. The MVFF for linear time-invariant systems was proposed in [22]. An extended version of the MVFF for time-varying systems was proposed in [14]. The estimation error covariance of the MVFF was computed in [12]. Readers can find the detailed derivation of the MVFF in [14], [22], and [23].

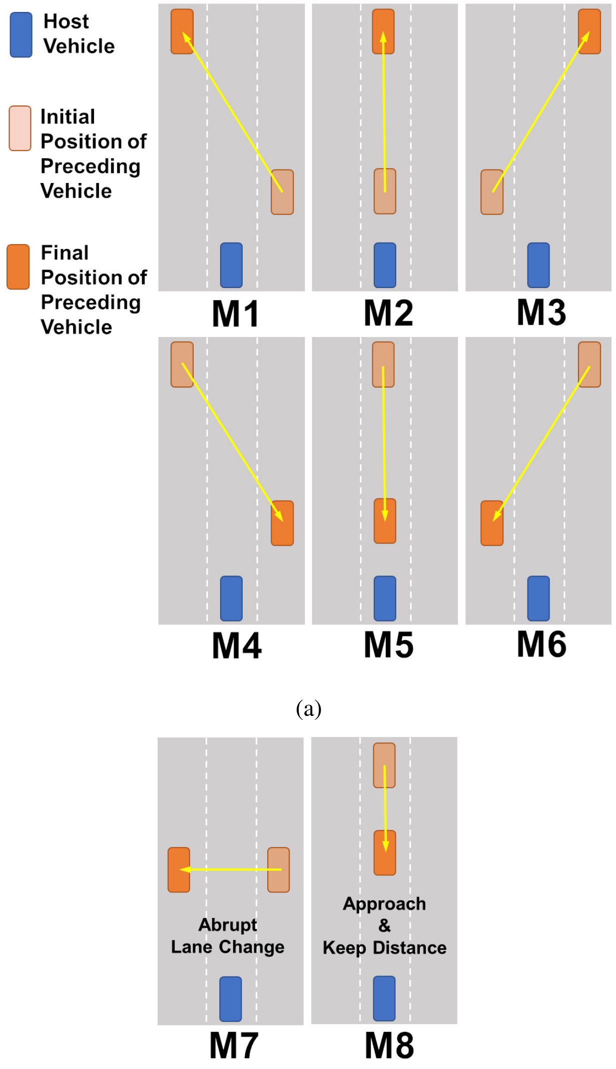
When there is no measurement in the validation region at time k, the HPFF produces $\hat{\mathbf{x}}_k^*$ and $\mathbf{P}_k^*$ using the assisting MVFF. Moreover, $\hat{\mathbf{x}}_k^*$ becomes the state estimate at time k, and the main filter (PDAF) is reset as $\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^*$ and $\mathbf{P}_k^+ = \mathbf{P}_k^*$. At the next time step $k+1$, the main PDAF runs with $\hat{\mathbf{x}}_k^+$ and $\mathbf{P}_k^+$ The overall algorithm of the HPFF is summarized in *Algorithm* 1.

Algorithm 1 HPFF

```
Input: $\hat{\mathbf{x}}_{k-1}^+$, $\mathbf{P}_{k-1}^+$
Output: $\hat{\mathbf{x}}_k^+$, $\mathbf{P}_k^+$
Perform the time update process using (11) and obtain $\hat{\mathbf{x}}_k^-$ and $\mathbf{P}_k^-$.
Convert the range/bearing measurements into Cartesian measurements using (4) and (5) and obtain $\mathcal{Z}_k = \{\mathbf{z}_k(i)\}_{i=1}^{m(k)}$ and $\mathcal{R}_k = \{\mathbf{R}_k(i)\}_{i=1}^{m(k)}$.
Perform the measurement validation process:
$\hat{\mathbf{z}}_k = \mathbf{C}\hat{\mathbf{x}}_k^-$
for $i = 1, 2, \ldots, m(k)$ do
    $\mathbf{S}_k(i) = \mathbf{C}\mathbf{P}_k^-\mathbf{C}^T + \mathbf{R}_k(i)$
    $\nu_k(i) = (\mathbf{z}_k(i) - \hat{\mathbf{z}}_k)^T\mathbf{S}_k(i)^{-1}(\mathbf{z}_k(i) - \hat{\mathbf{z}}_k)$
    if $\nu_k(i) \leq \gamma$ then
        Save $\mathbf{z}_k(i)$ as a validated measurement.
    end if
end for
if $n(k) > 0$ then
    for $i = 1, 2, \ldots, n(k)$ do
        Compute the association probability using (14)–(16).
        Perform the measurement update process using (17)–(22) and obtain $\hat{\mathbf{x}}_k^+$ and $\mathbf{P}_k^+$.
    end for
else
    Operate the MVFF using (26) and obtain $\hat{\mathbf{x}}_k^*$ and $\mathbf{P}_k^*$.
    Reset the PDAF as follows:
    $\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^*$
    $\mathbf{P}_k^+ = \mathbf{P}_k^*$
end if
$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^+$
Obtain and save the nearest neighbor measurement at time $k$ for the assisting FIR filtering in the future.
```

IV. SIMULATION RESULTS

This section discusses the simulation results to evaluate the performance of the proposed HPFF. In the simulation, the position and velocity of a preceding vehicle were tracked using both the HPFF and the conventional PDAF. Various motions of the preceding vehicle were considered, which are illustrated in Fig. 3 and are numbered as M1, M2, …, M8. Fig. 3 (a) shows the constant velocity motions. The initial positions ((x, y)) of the preceding vehicle for M1–M6 were set to $(10, 3.5)$, $(10, 0)$, $(10, -3.5)$, $(100, -3.5)$, $(100, 0)$, $(100, 3.5)$, respectively, all in meters. Fig. 3 (b) shows the abruptly changing motions. In the case of M7, the preceding vehicle was initially located at the third lane, and it abruptly moved to the first lane. The width of a lane was assumed to be 3.5 m. In the case of M8, the distance between the host and preceding vehicles decreased, which might have been due to the acceleration of the host vehicle or the deceleration of the


Host Vehicle
Initial Position of Preceding Vehicle
Final Position of Preceding Vehicle
M1
M2
M3
M4
M5
M6
Abrupt Lane Change
Approach & Keep Distance
M7
M8

(a)

(b)

FIGURE 3. Scenarios of preceding vehicle motion: (a) constant velocity motions, and (b) abruptly changing motions.

preceding vehicle. The relative speed was adjusted for a short time interval, and the distance between the two vehicles was maintained afterward.

Simulations were conducted for the eight motions described above, two clutter conditions (moderate and heavy clutters), and six detection probabilities. The preceding vehicle tracking was conducted for 19.5 s. The sampling interval was set as $T = 0.3\,s$. The automotive radar reported range/bearing measurements with noise covariances $\sigma_r = 0.25\,m$ and $\sigma_\theta = 1.5^\circ$ [20], [24]. The clutter density λ was set to 1.0×10^{-2} and 1.0×10^{-1} for the moderate and heavy clutter environments, respectively. False measurements (clutters) were generated independently and were uniformly distributed in a square centered at the correct measurement. The area of the square was $V = N_F/\lambda$, and N_F is the number of false measurements, which was computed as

$$N_F = [10V_k\lambda + 1]^-, \tag{27}$$

$$V_k = \pi\gamma|S_k|^{1/2}, \tag{28}$$

where the notation $[\cdot]^-$ means rounded down to the nearest integer. In (28), S_k was obtained using the standard KF assuming a clean (no clutters) environment [6].

The default value of detection probability P_D was set to 0.95. The gate probability was set as $P_G = 0.99$. The horizon size of the MVFF was set as $N = 4$. The gate threshold was $\gamma = 9.21$.

The initialization of both the PDAF and HPFF was performed with the two-point differencing method assuming a clean environment [6]. Because the first and second measurements were used for the initialization, the filters produced estimated positions from the third measurement (i.e., after 0.9 s). For evaluating the filter performance, the position error at time k was computed as follows:

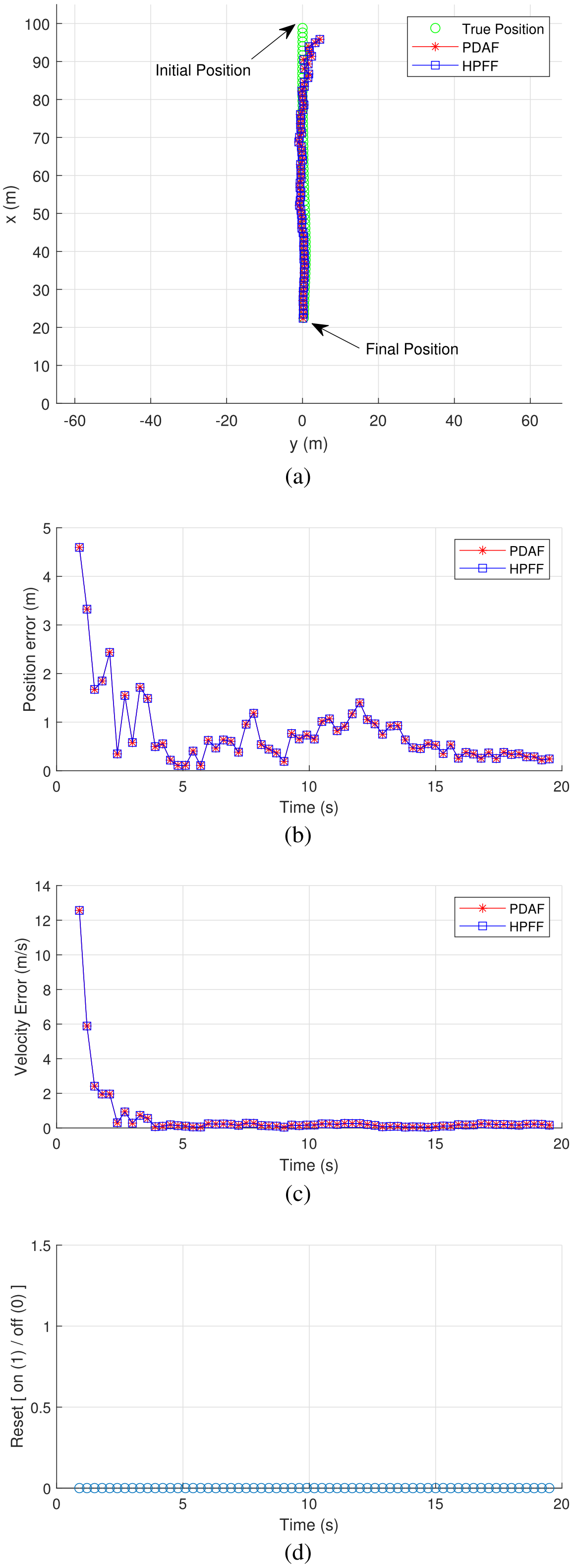
$$\sqrt{(x_k - \hat{x}_k)^2 + (y_k - \hat{y}_k)^2}, \tag{29}$$

where (x_k, y_k) and $(\hat{x}_k, \hat{y}_k)$ are the true and estimated positions, respectively.

Fig. 4 shows the simulation results of a successful track case obtained with the M5 motion scenario in a moderate clutter environment. In this case, the PDAF successfully tracked the target (preceding vehicle), and reset by the assisting FIR filter was not conducted. Thus, the HPFF operated as if it was a pure PDAF. Fig. 4 (a) shows the true and estimated positions. The estimated positions initially deviated from the true positions because radars typically exhibit large errors in bearing measurement at a long range. Figs. 4 (b) and 4 (c) show the position and velocity errors as functions of time. The PDAF and HPFF exhibited the same errors, which gradually decreased because the distance between the target and radar decreased. Fig. 4 (d) shows the reset timing diagram that indicates the instance when the assisting FIR filter operated and the PDAF in the HPFF was reset. There was no reset because the main PDAF in the HPFF successfully tracked the target.

Fig. 5 shows the simulation results obtained in the case of M6 in a moderate clutter environment. In this case, the PDAF failed in tracking the target, but the HPFF successfully tracked the target because of the reset by the assisting FIR filter. Fig. 5 (d) shows that the reset was conducted three times. Lost tracks like this case rarely occurred in the M1–M6 cases in a moderate clutter environment.

Fig. 6 shows the simulation results obtained with the M7 motion scenario in a moderate clutter environment. The preceding vehicle started to change the lane at 4.8 s and finished the lane change at 6.6 s. During this time interval, the position errors of the PDAF sharply increased to approximately 7 m, which is the same as the distance that travelled by the preceding vehicle. This means that the position estimated by the PDAF remained in the third lane. The PDAF did not



(a)

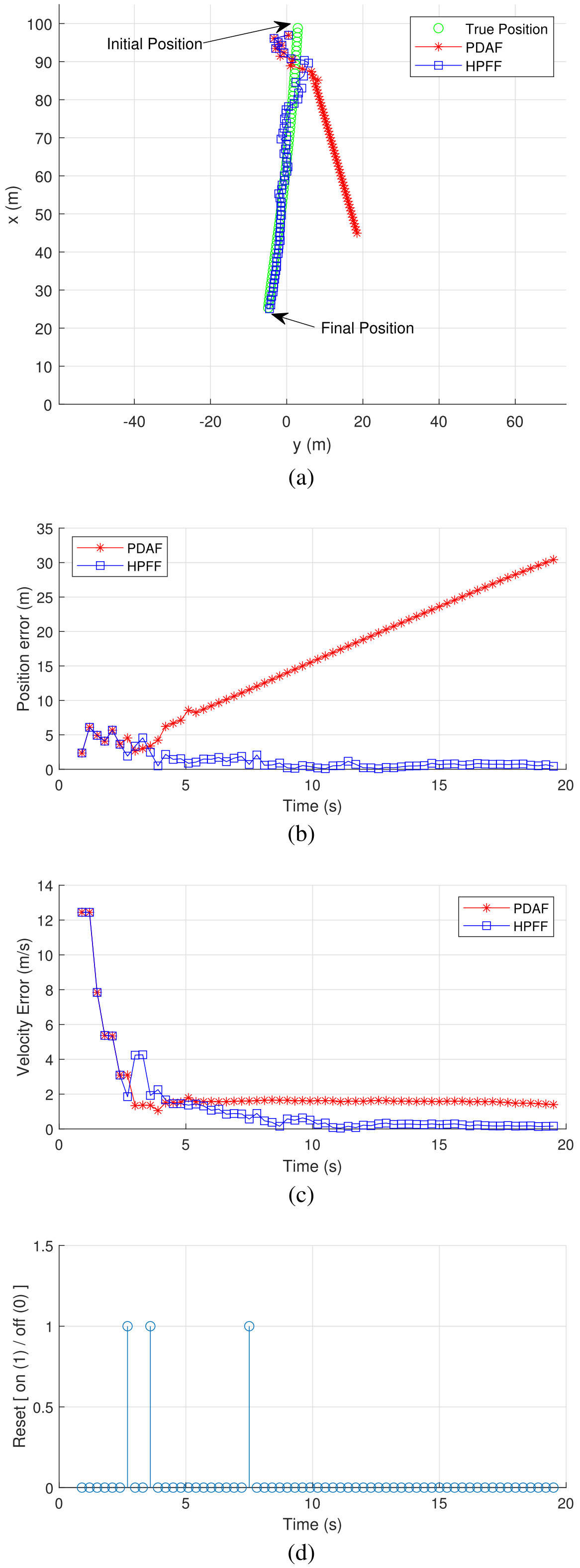
(b)

(c)

(d)

FIGURE 4. Simulation results with M5 motion scenario in moderate clutter environment: (a) true and estimated positions, (b) RMS position error, (c) RMS velocity error, and (d) reset timing.

recognize the lane change and lost the target vehicle. In contrast, the HPFF could track the preceding vehicle during the lane change. This was due to the assistance of the FIR filter.



(a)

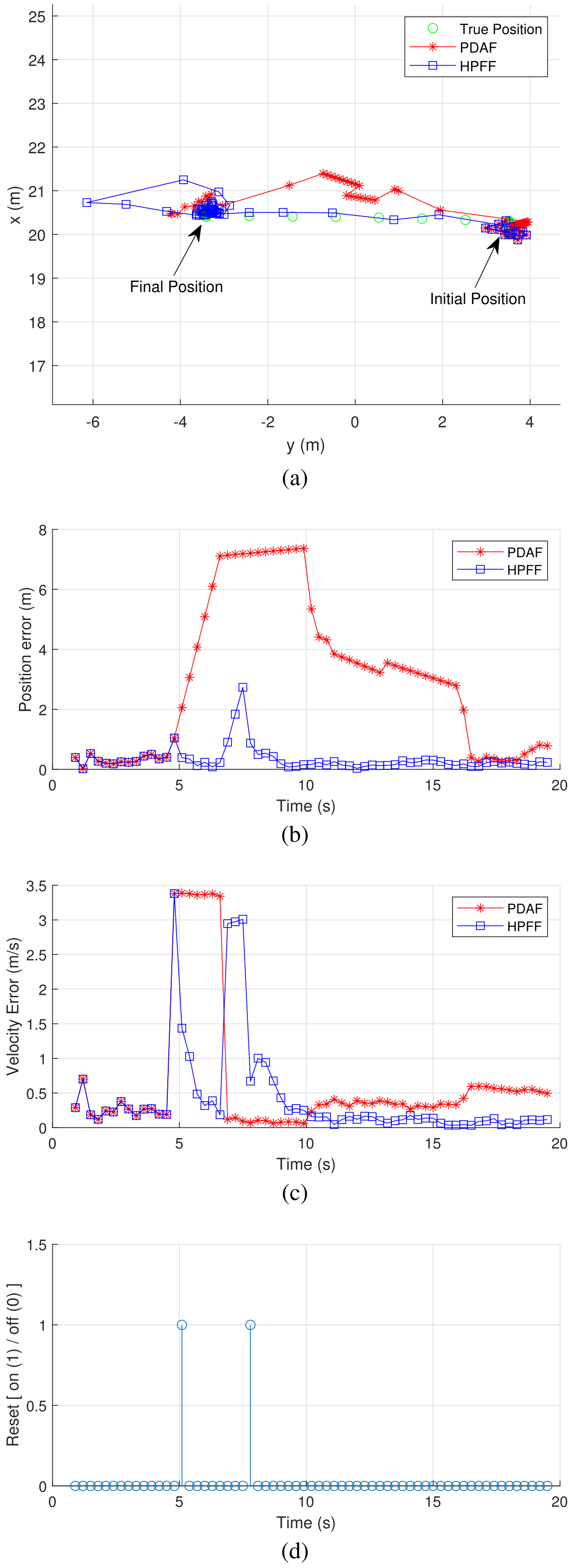
(b)

(c)

(d)

FIGURE 5. Simulation results with M6 motion scenario in moderate clutter environment): (a) true and estimated positions, (b) RMS position error, (c) RMS velocity error, and (d) reset timing.

Fig. 6 (d) shows that the reset occurred at the start and finish times of the lane change. In these two instances, the changes in the motion of the target were significant.



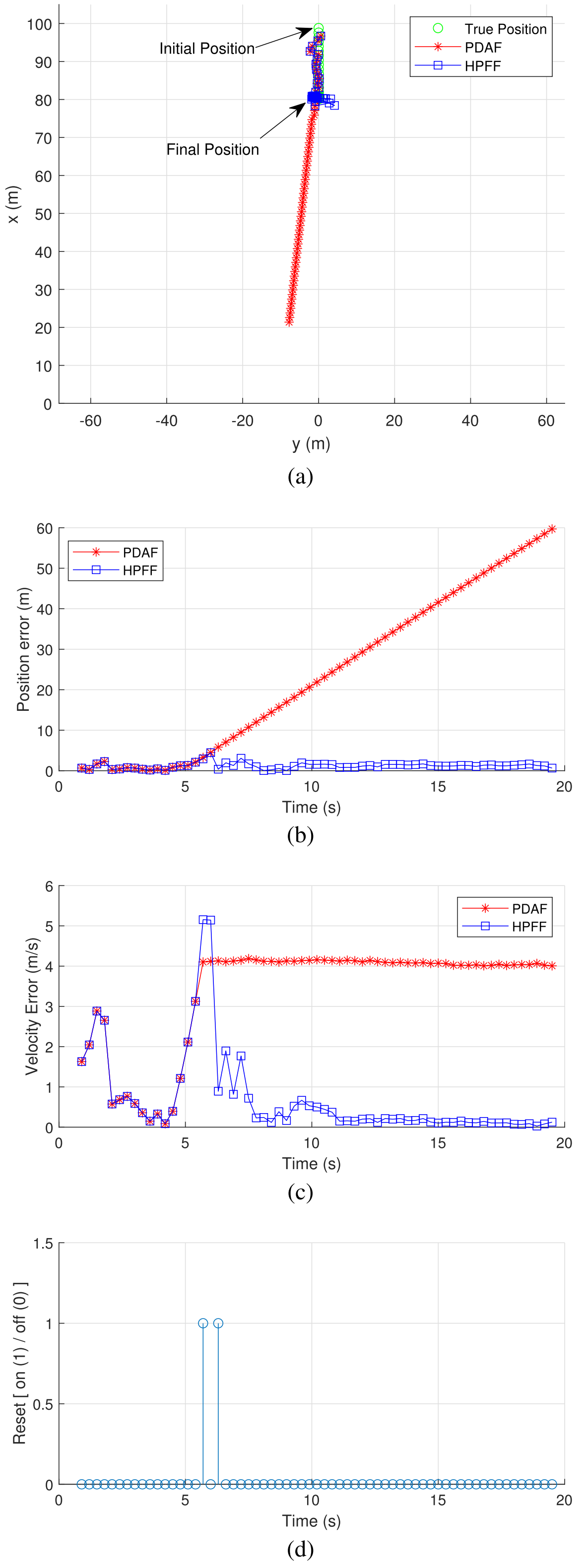
(a)

(b)

(c)

(d)

FIGURE 6. Simulation results with M7 motion scenario in moderate clutter environment): (a) true and estimated positions, (b) RMS position error, (c) RMS velocity error, and (d) reset timing.



(a)

(b)

(c)

(d)

FIGURE 7. Simulation results with M8 motion scenario in moderate clutter environment: (a) true and estimated positions, (b) RMS position error, (c) RMS velocity error, and (d) reset timing.

Fig. 7 shows the simulation results obtained in the case of M8 in a moderate clutter environment. The distance between the two vehicles decreased constantly for 4.5 *s*. The relative velocity decreased to zero for 1.2 *s*, and then the distance maintained between the vehicles was approximately 80 *m*, as shown in Fig. 7 (a). In Fig. 7 (b), the position error of

TABLE 2. Monte carlo simulation results with M1–M6 motions in a moderate clutter environment.

Performance	M1		M2		M3		M4		M5		M6	
Metric	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF
RTAMS position error (m)	0.56	0.52	0.57	0.52	0.59	0.53	3.23	0.90	3.34	0.90	3.33	0.91
RTAMS velocity error (m/s)	0.17	0.29	0.17	0.28	0.17	0.29	0.73	0.68	0.74	0.67	0.74	0.68
Percentage of lost track	0.6	0.0	0.6	0.0	0.7	0.0	3.2	0.0	3.1	0.0	3.2	0.0

TABLE 3. Monte carlo simulation results with M1–M6 motions in a heavy clutter environment.

Performance	M1		M2		M3		M4		M5		M6	
Metric	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF
RTAMS position error (m)	1.39	0.70	1.48	0.70	1.38	0.71	3.08	1.68	2.71	1.64	2.96	1.68
RTAMS velocity error (m/s)	0.27	0.31	0.28	0.31	0.27	0.31	0.81	0.91	0.76	0.89	0.79	0.91
Percentage of lost track	5.7	2.2	6.1	2.3	5.9	2.3	18.0	3.5	17.4	3.1	18.1	3.1

TABLE 4. Monte carlo simulation results for a range of detection probabilities.

Performance	$P_D = 0.95$		$P_D = 0.90$		$P_D = 0.85$		$P_D = 0.80$		$P_D = 0.75$		$P_D = 0.70$	
Metric	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF	PDAF	HPFF
RTAMS position error (m)	3.34	0.90	3.96	1.04	4.02	1.17	4.24	1.32	4.45	1.46	4.73	1.62
RTAMS velocity error (m/s)	0.74	0.67	0.82	0.83	0.86	0.97	0.90	1.12	0.95	1.25	1.01	1.39
Percentage of lost track	3.1	0.0	4.1	0.0	4.3	0.0	5.3	0.0	5.6	0.0	6.8	0.0

the PDAF increased sharply because of the abrupt motion change. Fig. 7 (c) shows that the relative velocity estimated by the PDAF remained 4 m/s, which means that the PDAF did not recognize the changes in the relative velocity. In contrast, the HPFF successfully tracked the target vehicle because of the reset by the assisting FIR filter. In Fig. 7 (d), the reset in the HPFF occurred three times.

During the simulation, the PDAF sometimes lost a target vehicle when it suddenly changed the motion. However, the HPFF successfully tracked the target because the main filter (i.e., PDAF) was reset by the assisting FIR filter (i.e., MVFF). The reset was conducted when the main filter lost the target, and thus there was no measurements in the validation region.

Tables 2–6 show the Monte Carlo (MC) simulation results, in which the root time averaged mean square (RTAMS) errors (units in meters) and the percentage of the lost tracks were computed. The RTAMS position error is defined as

$$\sqrt{\sum_{k=3}^{t_{max}}\sum_{i=1}^{M}[(x_k^i-\hat{x}_k^i)^2+(y_k^i-\hat{y}_k^i)^2]}, \tag{30}$$

where i indicates the i-th MC run, and the RTAMS velocity error can be obtain by replacing the 2D positions with the 2D velocities. The final time step t_{max} was set to 65, and the number of MC runs M was set to $10,000$. A track was considered lost when the correct measurement was not in the validation region for at least the last 20 sampling times [6]. We computed the percentage of lost tracks among the $10,000$ MC runs.

Tables 2 and 3 are the MC simulation results when the preceding vehicle moves with constant velocity. Table 2 shows that the PDAF and the HPFF exhibit similar performances under moderate conditions. In Table 3, the performance of both filters deteriorated in the heavy clutter environment, but the HPFF exhibited better reliability than the PDAF in terms of the percentage of lost tracks.

TABLE 5. Monte carlo simulation results with M7 and M8 motions in a moderate clutter environment.

Performance	M7		M8	
Metric	PDAF	HPFF	PDAF	HPFF
RTAMS position error (m)	3.35	0.67	24.97	2.21
RTAMS velocity error (m/s)	0.66	0.85	3.61	2.04
Percentage of lost track	35.6	0.0	99.8	1.2

TABLE 6. Monte carlo simulation results with M7 and M8 motions in a heavy clutter environment.

Performance	M7		M8	
Metric	PDAF	HPFF	PDAF	HPFF
RTAMS position error (m)	3.87	1.47	19.32	4.69
RTAMS velocity error (m/s)	0.87	1.42	3.22	2.53
Percentage of lost track	40.5	4.4	99.8	41.2

Table 4 summarizes the MC simulation results for a range of P_D. As P_D decreased, the probability that the radar does not detect the true target increased. Thus, the tracking performance worsend as P_D decreased, which can be confirmed from Table 4. The HPFF exhibited better performance than the PDAF even when P_D decreased.

Tables 5 and 6 are the MC simulation results when the preceding vehicle abruptly changes the motion. Under these severe conditions, the PDAF exhibited much larger lost tracks than the HPFF. In the M8 motion scenario, the PDAF lost the target with a probability 99.8 %. However, the HPFF's percentage of lost track was only 1.2 % in a moderate clutter environment. This demonstrates that the HPFF is much more reliable than the conventional PDAF under severe conditions.

V. CONCLUSION

In this paper, the HPFF algorithm was proposed for tracking a preceding vehicle using automotive radars. The HPFF is a combination of a main filter, the PDAF, and an assisting filer, the FIR filter. Although the PDAF is successful in normal situations, it can lose a target under severe conditions. The proposed HPFF recovers the main PDAF after failure by resetting it using the assisting FIR filter. Thus, the HPFF can provide reliable preceding vehicle tracking, which was demonstrated via simulations. The HPFF proposed in this study is suitable only for single-target tracking, and in the future we will develop a new HPFF for multiple-target tracking.

REFERENCES

[1] C. Waldschmidt, J. Hasch, and W. Menzel, "Automotive radar—From first efforts to future systems," *IEEE J. Microw.*, vol. 1, no. 1, pp. 135–148, Jan. 2021.

[2] H. Liu, C. Huang, L. Gan, Y. Zhou, and T.-K. Truong, "Clutter reduction and target tracking in through-the-wall radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 486–499, Jan. 2020.

[3] C. Cao, J. Zhang, J. Meng, X. Zhang, and X. Mao, "Clutter suppression and target tracking by the low-rank representation for airborne maritime surveillance radar," *IEEE Access*, vol. 8, pp. 160774–160789, 2020.

[4] J. Liu, H. Meng, H. Zhang, and X. Wang, "Radar sea clutter suppression and target indication with a spatial tracking filter," *Tsinghua Sci. Technol.*, vol. 15, no. 2, pp. 228–234, Apr. 2010.

[5] Y. Bar-Shalom and E. Tse, "Tracking in a cluttered environment with probabilistic data association," *Automatica*, vol. 11, no. 5, pp. 451–460, Jun. 1975.

[6] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. Bradford, U.K.: YBS Publishing, 2011.

[7] S. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Norwood, MA, USA: Arctech House, 1999.

[8] S. Zhao, Y. S. Shmaliy, J. A. Andrade-Lucio, and F. Liu, "Multipass optimal FIR filtering for processes with unknown initial states and temporary mismatches," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5360–5368, Aug. 2021.

[9] S. Zhao, Y. S. Shmaliy, and F. Liu, "Optimal FIR filter for discrete-time LTV systems and fast iterative algorithm," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 68, no. 4, pp. 1527–1531, Apr. 2021.

[10] S. Zhao, Y. S. Shmaliy, C. K. Ahn, and F. Liu, "Self-tuning unbiased finite impulse response filtering algorithm for processes with unknown measurement noise covariance," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 3, pp. 1372–1379, May 2021.

[11] J. M. Pak, C. K. Ahn, P. Shi, Y. S. Shmaliy, and M. T. Lim, "Distributed hybrid particle/FIR filtering for mitigating NLOS effects in TOA-based localization using wireless sensor networks," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5182–5191, Jun. 2017.

[12] J. M. Pak, C. K. Ahn, Y. S. Shmaliy, and M. T. Lim, "Improving reliability of particle filter-based localization in wireless sensor networks via hybrid particle/FIR filtering," *IEEE Trans. Ind. Informat.*, vol. 11, no. 5, pp. 1089–1098, Oct. 2015.

[13] J. M. Pak, C. K. Ahn, Y. S. Shmaliy, P. Shi, and M. T. Lim, "Switching extensible FIR filter bank for adaptive horizon state estimation with application," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 3, pp. 1052–1058, May 2016.

[14] J. M. Pak, C. K. Ahn, M. T. Lim, and M. K. Song, "Horizon group shift FIR filter: Alternative nonlinear filter using finite recent measurements," *Measurement*, vol. 57, pp. 33–45, Nov. 2014.

[15] Y. S. Shmaliy, "Suboptimal FIR filtering of nonlinear models in additive white Gaussian noise," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5519–5527, Oct. 2012.

[16] Y. S. Shmaliy, "Linear optimal FIR estimation of discrete time-invariant state-space models," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3086–3096, Jun. 2010.

[17] Y. S. Shmaliy, "Unbiased FIR filtering of discrete-time polynomial state-space models," *IEEE Trans. Signal Process.*, vol. 57, no. 4, pp. 1241–1249, Apr. 2009.

[18] J. M. Pak, P. S. Kim, S. H. You, S. S. Lee, and M. K. Song, "Extended least square unbiased FIR filter for target tracking using the constant velocity motion model," *Int. J. Control, Autom. Syst.*, vol. 15, no. 2, pp. 947–951, Apr. 2017.

[19] J. M. Pak, C. K. Ahn, Y. S. Shmaliy, P. Shi, and M. T. Lim, "Accurate and reliable human localization using composite particle/FIR filtering," *IEEE Trans. Human-Mach. Syst.*, vol. 47, no. 3, pp. 332–342, Jun. 2017.

[20] S.-K. Han, W.-S. Ra, I.-H. Whang, and J. B. Park, "Linear recursive automotive target tracking filter for advanced collision warning systems," *Appl. Math. Inf. Sci.*, vol. 8, no. 3, pp. 1145–1151, May 2014.

[21] J. A. Mcgee, T. E. Luginbuhl, J. H. Dibiase, and P. L. Ainsleigh, "A modified PDAF algorithm for measurement covariances that vary within a time scan," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 1, pp. 906–912, Jan. 2012.

[22] W. H. Kwon and S. Han, *Receding Horizon Control: Model Predictive Control for State Models*. London, U.K.: Springer-Verlag, 2005.

[23] J. M. Pak, S. Y. Yoo, M. T. Lim, and M. K. Song, "Weighted average extended FIR filter bank to manage the horizon size in nonlinear FIR filtering," *Int. J. Control, Autom. Syst.*, vol. 13, no. 1, pp. 138–145, Feb. 2015.

[24] S.-K. Han, W.-S. Ra, I.-H. Whang, and J. B. Park, "Gaussian mixture approach to decision making for automotive collision warning systems," *Int. J. Control, Autom. Syst.*, vol. 13, no. 5, pp. 1182–1192, Jul. 2015.

JUNG MIN PAK (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea University, Seoul, Republic of Korea, in 2006, 2008, and 2015, respectively. From 2016 to 2020, he was an Assistant Professor with the Department of Electrical Engineering, Wonkwang University, Iksan-si, Republic of Korea, where he is currently an Associate Professor with the Department of Electrical Engineering. His research interests include state estimation, target tracking, and localization.