Article

Finite Impulse Response Filter-Based Track Formation for Preceding Vehicle Tracking Using Automotive Radars

Jung Min Pak

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

Abstract: Automotive radars, which are used for preceding vehicle tracking, have attracted significant attention in recent years. However, the false measurements that occur in cluttered roadways hinders the tracking process in vehicles; thus, it is essential to develop automotive radar systems that are robust against false measurements. This study proposed a novel track formation algorithm to initialize the preceding vehicle tracking in automotive radar systems. The proposed algorithm is based on finite impulse response filtering, and exhibited significantly higher accuracy in highly cluttered environments than a conventional track formation algorithm. The excellent performance of the proposed algorithm was demonstrated using extensive simulations under real conditions.

Keywords: automotive radars; finite impulse response (FIR) filter; preceding vehicle tracking; track formation

1. Introduction

An automotive radar is a system for detecting obstacles around a host vehicle on the road using microwaves. Currently, automotive radars have attracted increasing attention, and have been applied for tracking the position and velocity of preceding vehicles [1–8]. As many vehicles on the road are equipped with automotive radars, radar signals emitted from the radar of other vehicles may be detected by the receiver of the host vehicle. These signals result in false measurements, which hinder the detection and tracking of true targets [4,5]. In addition, the scattered reflection of radar signals in cluttered road environments may cause increasing false measurements [7]. Thus, it is essential to develop or improve automotive radar systems that are robust against the false measurements [8].

Received radar signals may or may not include the signals from a true target. Thus, it is essential to develop a process that can determine the presence of a signal from a true target in the received signals, and this process in known as the detection process. Thresholding is a representative and simple detection process, and various signal processing techniques, including artificial intelligence algorithms, have been utilized for the detection process [1,3,9,10]. However, because the detected measurements include false measurements, the number of detected measurements can be larger than the actual number of targets. After the detected measurements are obtained, a tracking process is performed.

Various tracking algorithms have been investigated and developed. For example, the probabilistic data association filter (PDAF) [11] has been developed as a representative algorithm for tracking a single target in a clutter [12]. In addition, joint probabilistic data association [13], multiple hypothesis tracking [14], and probability hypothesis density filter [15] have been developed for tracking multiple targets. Generally, tracking algorithms are based on state estimation (also known as stochastic filtering), which requires an initial state estimate and estimation error covariance. The process of generating the initial estimate using the detected measurements is known as the track formation. However, a large number of false measurements in highly cluttered environments hinders the track formation process [12].

This study proposed a novel track formation algorithm to address the problem of tracking a single preceding vehicle using automotive radars. Currently, two representative...
conventional algorithms are available for this problem [12]: The cascaded logic track formation, which is a heuristic algorithm, and the multiple model track formation (MMTF), which is an advanced algorithm based on the interacting multiple model (IMM) method [12]. These methods utilize the two-point differencing method, which computes the initial estimate of the velocity by differencing two successive measurements. In contrast, the algorithm proposed in this study is based on finite impulse response (FIR) filtering, which has recently attracted attention from researchers [16–24]. FIR filters are batch-processing filters, which can produce state estimate and error covariance using recent finite measurements, without filter initialization. Inspired by these characteristics of the FIR filter, in this study, we designed a novel FIR filter-based track formation (FFTF) algorithm. The FFTF algorithm proposed in this study exhibited a significantly higher accuracy than the conventional MMTF under highly heavy clutter conditions. Moreover, despite changes in the relative distance and velocities between vehicles, the FFTF provided consistent track formation. The excellent performance of the FFTF was demonstrated using extensive simulations that considered various real-world conditions.

This paper is organized as follows. Section 2 introduces the track formation problem for a single preceding vehicle using automotive radars in cluttered environments. Section 3 describes the MMTF, and Section 4 proposes the FFTF algorithm. Section 5 presents simulation results to demonstrate the performance of the FFTF, and the conclusions are stated in Section 6.

2. Preceding Vehicle Tracking Using Automotive Radars in Clutters

Generally, an automotive radar for preceding vehicle tracking is installed in the middle front of the host vehicle, as shown in Figure 1 [20]. The automotive radar emits millimeter waves with a certain field of view (FOV), after which the radar waves scatter and are reflected from various objects on a road. The reflected signals from nontarget objects are referred to as clutters, which hinder the tracking of preceding vehicle. Moreover, owing to the widespread use of automotive radars, the emission of signals from the radars of other vehicles results in the formation of increased clutters. Therefore, this study aimed to estimate the position and velocity of a preceding target vehicle in highly-heavy clutter environment. The position of the preceding vehicle is represented in a two-dimensional (2D) Cartesian coordinate system, as shown in Figure 1. The origin was located on the position of the radar, and the x-axis was aligned with the longitudinal direction of the host vehicle. The radar measured the range, \( r \), and bearing, \( \theta \), between the host and preceding vehicles, and the measurements included noise. The measured range and bearing, \( r_m \) and \( \theta_m \), can be represented as

\[
\begin{align*}
    r_m &= r + \bar{\nu}_r, \\
    \theta_m &= \theta + \bar{\nu}_\theta,
\end{align*}
\]

where \( \bar{\nu}_r \) and \( \bar{\nu}_\theta \) are the measurement noises, which are assumed to be the independent and zero mean white Gaussian noises with covariances \( \sigma_r^2 \) and \( \sigma_\theta^2 \), respectively, [12].

To obtain the preceding vehicle positions in the Cartesian coordinate, the range and bearing measurements were converted as follows:

\[
\begin{align*}
    x_m &= r_m \cos(\theta_m), \\
    y_m &= r_m \sin(\theta_m),
\end{align*}
\]

where \( x_m \) and \( y_m \) are the converted measurements, and they can be also represented as:

\[
\begin{align*}
    x_m &= x + \bar{\nu}_x, \\
    y_m &= y + \bar{\nu}_y.
\end{align*}
\]
where $\bar{v}_x$ and $\bar{v}_y$ are the measurement noises in the Cartesian coordinate. The measurement vector was defined as $z = [x_m \ y_m]^T$, and the noise covariance of $z$ was obtained as follows [12,20]:

$$
R = \begin{bmatrix}
R_{11} & R_{12} \\
R_{12} & R_{22}
\end{bmatrix},
$$

$$
R_{11} = r_m^2 \sigma_x^2 \sin^2(\theta_m) + \sigma_r^2 \cos^2(\theta_m),
$$

$$
R_{22} = r_m^2 \sigma_y^2 \cos^2(\theta_m) + \sigma_r^2 \sin^2(\theta_m),
$$

$$
R_{12} = (\sigma_r^2 - r_m^2 \sigma_x^2) \sin(\theta_m) \cos(\theta_m).
$$

(4)

Equation (4) indicates that $R$ depends on the measurements of $r_m$ and $\theta_m$. Thus, $R$ varied with time, which is because $r_m$ and $\theta_m$ varied with time owing to random noises.

Typically, stochastic filters (state estimators) are used in radar-based target tracking to reduce the effects of the measurement noise. The stochastic filtering (state estimation) requires state-space system models that are composed of motion and measurement models. In the preceding vehicle tracking, the state of the target is defined as

$$
x = [x \ v_x \ y \ v_y]^T,
$$

where $v_x$ and $v_y$ are the 2D velocities along the $x$- and $y$-axes. If the state vector at a discrete time step $k$ is defined as $x(k) = [x(k) \ v_x(k) \ y(k) \ v_y(k)]^T$, the motion model can be represented

$$
x(k+1) = Fx(k) + Gw(k),
$$

$$
F = \begin{bmatrix}
1 & T & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & T \\
0 & 0 & 0 & 1
\end{bmatrix},
$$

$$
G = \begin{bmatrix}
T^2/2 & 0 \\
0 & T \\
0 & T^2/2 \\
0 & T
\end{bmatrix},
$$

(5)

where $T$ is the sampling interval and $w(k)$ is the white Gaussian process noise that represents the acceleration. Therefore, the motion model (5) is referred to as the discrete white noise acceleration (DWNA) model. The covariance of $w(k)$ can be represented as $Q(k) = \sigma_w^2 I_2$, where $I_2$ is the $2 \times 2$ identity matrix.

Figure 1. Automotive radar geometry.
The measurement model expresses the relationship between measurement and state. If the measurement vector at time \( k \) is defined as \( z(k) = [x_m(k) y_m(k)]^T \), the measurement model can be represented as:

\[
    z(k) = Hx(k) + \nu(k),
\]

\[
    H = \begin{bmatrix}
        1 & 0 & 0 & 0 \\
        0 & 0 & 1 & 0
    \end{bmatrix},
\]

where \( \nu(k) \) is the measurement noise vector, and its covariance is \( R(k) \). As previously mentioned, \( R(k) \) varies with time, and should be computed at each time step using (4).

PDAF is one of the most renowned stochastic filter for single target tracking in clutter. PDAF is the combination of the probabilistic data association (PDA) algorithm and the Kalman filter (KF). In this filter, PDA handles the clutter, and KF reduces the effects of measurement noises. KF recursively updates the state estimate and the estimation error covariance at each time step, indicating that it requires an initial state estimate and covariance. KF begins with track initiation, which involves the generation of preliminary tracks, where “track” indicates a sequence (history) of measurements or the estimation results (state estimate and covariance) obtained from the sequence of measurements obtained at times \( k \) is denoted as \( m_1(k) \) and \( m_2(k) \). The number of measurements obtained at times \( k \) is denoted as \( m(k) \), \( m_1(k) \times m_2(k) \) pairs of measurements can be obtained. However, the number of measurement pairs in highly-heavy clutter environment is very large (more than a thousand). Among the large number of measurement pairs, unreasonable pairs that exceed a maximum possible velocity are discarded through the gating technique. The gate is defined as a rectangle with an area of

\[
    V_{\text{rect}} = 2(v_{x,\text{max}} \times T + 2\sqrt{R_{11}}) | 2(v_{y,\text{max}} \times T + 2\sqrt{R_{22}}),
\]

where \( v_{x,\text{max}} \) and \( v_{y,\text{max}} \) are the maximum possible velocities along with the x- and y-axes, respectively. The measurement pairs remaining after the gating process are known as the preliminary tracks, where “track” indicates a sequence (history) of measurements or the estimation results (state estimate and covariance) obtained from the sequence of

3. Multiple Model Track Formation and Problem Formulation

In this section, we introduce a state-of-the-art track formation algorithm known as the MMTF. The MMTF begins with track initiation, which involves the generation of preliminary tracks via the combination of measurements at times \( k = 1 \) and \( k = 2 \). If the number of measurements obtained at times \( k \) is denoted as \( m(k) \), \( m_1(k) \times m_2(k) \) pairs of measurements can be obtained. However, the number of measurement pairs in highly-heavy clutter environment is very large (more than a thousand). Among the large number of measurement pairs, unreasonable pairs that exceed a maximum possible velocity are discarded through the gating technique. The gate is defined as a rectangle with an area of

\[
    V_{\text{rect}} = 2(v_{x,\text{max}} \times T + 2\sqrt{R_{11}}) | 2(v_{y,\text{max}} \times T + 2\sqrt{R_{22}}),
\]

where \( v_{x,\text{max}} \) and \( v_{y,\text{max}} \) are the maximum possible velocities along with the x- and y-axes, respectively. The measurement pairs remaining after the gating process are known as the preliminary tracks, where “track” indicates a sequence (history) of measurements or the estimation results (state estimate and covariance) obtained from the sequence of

In real situations, false measurements (clutters) hinders the initialization process. Track formation algorithm determines the initial information required to start the actual estimation starts at \( k = 1 \). In addition, \( k = -1 \) indicates that it is one time step earlier than \( k = 0 \).

In this section, we introduce a state-of-the-art track formation algorithm known as the MMTF. The MMTF begins with track initiation, which involves the generation of preliminary tracks via the combination of measurements at times \( k = 1 \) and \( k = 2 \). If the number of measurements obtained at times \( k \) is denoted as \( m(k) \), \( m_1(k) \times m_2(k) \) pairs of measurements can be obtained. However, the number of measurement pairs in highly-heavy clutter environment is very large (more than a thousand). Among the large number of measurement pairs, unreasonable pairs that exceed a maximum possible velocity are discarded through the gating technique. The gate is defined as a rectangle with an area of

\[
    V_{\text{rect}} = 2(v_{x,\text{max}} \times T + 2\sqrt{R_{11}}) | 2(v_{y,\text{max}} \times T + 2\sqrt{R_{22}}),
\]

where \( v_{x,\text{max}} \) and \( v_{y,\text{max}} \) are the maximum possible velocities along with the x- and y-axes, respectively. The measurement pairs remaining after the gating process are known as the preliminary tracks, where “track” indicates a sequence (history) of measurements or the estimation results (state estimate and covariance) obtained from the sequence of

In real situations, false measurements (clutters) hinders the initialization process. Track formation algorithm determines the initial information required to start the actual estimation starts at \( k = 1 \). In addition, \( k = -1 \) indicates that it is one time step earlier than \( k = 0 \).

In this section, we introduce a state-of-the-art track formation algorithm known as the MMTF. The MMTF begins with track initiation, which involves the generation of preliminary tracks via the combination of measurements at times \( k = 1 \) and \( k = 2 \). If the number of measurements obtained at times \( k \) is denoted as \( m(k) \), \( m_1(k) \times m_2(k) \) pairs of measurements can be obtained. However, the number of measurement pairs in highly-heavy clutter environment is very large (more than a thousand). Among the large number of measurement pairs, unreasonable pairs that exceed a maximum possible velocity are discarded through the gating technique. The gate is defined as a rectangle with an area of

\[
    V_{\text{rect}} = 2(v_{x,\text{max}} \times T + 2\sqrt{R_{11}}) | 2(v_{y,\text{max}} \times T + 2\sqrt{R_{22}}),
\]

where \( v_{x,\text{max}} \) and \( v_{y,\text{max}} \) are the maximum possible velocities along with the x- and y-axes, respectively. The measurement pairs remaining after the gating process are known as the preliminary tracks, where “track” indicates a sequence (history) of measurements or the estimation results (state estimate and covariance) obtained from the sequence of
1. Compute the mixing probabilities:

\[ p_D^j = \begin{cases} 0, & \text{if } j = 1, \\ p_D, & \text{if } j = 2, \end{cases} \]  

(10)

where \( j \) is the model (mode) number. The processes of the IMMPDAF can be summarized as follows:

1. Compute the mixing probabilities:

\[ \mu_{ij}(k-1|k-1) = \frac{1}{c_j} p_{ij} \mu_i(k-1), \]  

(11)

\[ c_j = \sum_{i=1}^r p_{ij} \mu_i(k-1). \]  

(12)

2. Obtain an initial estimate and a covariance for the model-matched filter, \( j (j = 1, 2) \):

\[ \hat{x}^0_i(k-1|k-1) = \sum_{i=1}^r \hat{x}^0_i(k-1|k-1) \mu_{ij}(k-1|k-1), \]  

(13)

\[ P^0_i(k-1|k-1) = \sum_{i=1}^r \mu_{ij}(k-1|k-1) \left( P^i_i(k-1|k-1) + [\hat{x}^i(k-1|k-1) - \hat{x}^0_i(k-1|k-1)]^T \right). \]  

(14)

3. Perform a mode-conditioned filtering using PDAF (see Appendix A), and obtain \( \hat{x}^i(k|k) \) and \( P^i(k|k) \) for \( j = 1, 2 \).

4. Compute the likelihood functions:

\[ \Lambda_j[Z(k)] = \frac{1 - p_D^j}{V_m(k)} + \frac{m(k) p_D^j}{V_m(k)-1} \sum_{i=1}^m N[z_i(k); \hat{x}^i(k|k-1), S^i(k)], \]  

(15)

\[ V = c_n \gamma^n_{k/2} |S^i(k)|^{1/2}, \]  

(16)

\[ S^i(k) = H P^i(k|k) H^T + R(k), \]  

(17)

\[ \hat{x}^i(k|k-1) = H \hat{x}^i(k|k), \]  

(18)

where \( N[z_i(k); \hat{x}^i(k|k-1), S^i(k)] \) indicates the Gaussian probability density function with the mean \( \hat{x}^i(k|k-1) \) and covariance \( S^i(k) \); \( Z(k) \) is the set of validated measurements defined as \( Z(k) = \{ z_i(k) \}_{i=1}^m \); and \( z_i(k) \) is the \( i \)-th measurement obtained at time \( k \).

5. Update the mode probabilities:

\[ \mu_j(k) = \frac{1}{c} \Lambda_j[Z(k)] c_j, \]  

(19)

\[ c = \sum_{j=1}^r \Lambda_j[Z(k)] c_j. \]  

(20)
6. Obtain the estimate and covariance at time \( k \) by combination using:

\[
\hat{x}(k|k) = \sum_{j=1}^{r} \hat{x}'(k|k) \mu_j(k), \tag{21}
\]

\[
P(k|k) = \sum_{j=1}^{r} \mu_j(k) \left[ P'(k|k) + \left[ \hat{x}'(k|k) - \hat{x}(k|k) \right] \right] \\
\times \left[ \hat{x}'(k|k) - \hat{x}(k|k) \right]^T, \tag{22}
\]

where \( \hat{x}(k|k) \) and \( P(k|k) \) are the a posteriori (i.e., the measurements up to time \( k \) were used for estimation at time \( k \)) estimated state and estimation error covariance. In addition, \( \hat{x}(k|k-1) \) and \( P(k|k-1) \) indicate a priori (i.e., the measurements up to time \( k-1 \) were used for estimation at time \( k \)) estimated state and covariance.

At the initial time of preliminary tracking, \( k = 3 \), the mode probabilities were equally set as \( \mu_j(3) = 0.5 \). The transition probabilities from mode \( i \) to \( j \), denoted as \( p_{ij} \), were defined by the transition probability matrix as:

\[
\begin{bmatrix}
M_1 & M_2 \\
M_1 & 0.98 & 0.02 \\
M_2 & 0.02 & 0.98
\end{bmatrix} \tag{23}
\]

The mode probability of \( M_2 \), denoted as \( \mu_2(k) \), corresponds to the probability that a track belongs to a true target. Thus, \( \mu_2(k) \) is referred to as the true target probability (TTP).

During the preliminary tracking, tracks with extremely low TTP (e.g., smaller than 0.05) are discarded. This pruning results in a gradual decrease in the number of tentative tracks. In the preceding vehicle tracking where there is only one target vehicle, there should be only one track at the final time of preliminary tracking, \( k = N_W \). The track with the highest TTP is selected as the most reliable track.

As previously described, the MMTF is based on the two-point differencing method that is simple, but not accurate. Consequently, this significantly reduces the accuracy of the entire MMTF algorithm. Moreover, the TTP-based pruning cannot effectively reduce the number of tentative tracks in highly heavy clutter environments, which results in the computational burdening of the tracking algorithm.

4. FIR Filter-Based Track Formation

In this section, the novel track formation algorithm (i.e., the FFTF) proposed in this study is introduced. The most significant difference between the FFTF and the MMTF is in the track initiation process. Recursive filters including PDAF require an initial estimate and a covariance, which can be obtained using initialization methods (e.g., two-point differencing). However, FIR filters do not require the initialization, and can generate state estimate and covariances using recent finite number of measurements. Thus, FIR filters can be used for track initiation. Among various FIR filters, the minimum variance FIR filter (MVFF) was adopted in this study because it can produce not only the state estimate but also an estimation error covariance. The time-varying MVFF was proposed in [24] and was applied to localization problems in [21–23]. The equations of MVFF are written as follows:

\[
\hat{x}(k) = L \hat{Z}_N, \tag{24}
\]

\[
P(k) = K_N Q_N K_N^T + L R_N L^T, \tag{25}
\]

\[
L = J_N \begin{bmatrix} M_{1,1} & M_{1,2} \\
M_{1,2} & M_{2,2} \end{bmatrix}^{-1} \begin{bmatrix} \hat{C}_N^T \\
\hat{G}_N \end{bmatrix} R_N^{-1}, \tag{26}
\]
where $N$ is the horizon size of the MVFF, and $N$ should be greater than or equal to the dimension of the state vector. As reported in (24)–(34), the MVFF produces state estimate and covariance at time $k$ using the finite number of measurements on the horizon $[k - N, k - 1]$. Considering the dimension of the state vector in (5), the MVFF requires a minimum of four measurements.

Preliminary tracks for the FFTF are obtained in a similar manner to that of the MMTF. The only difference is that measurements from $k = 1$ to $k = 4$ are used for the FFTF. As shown in (24), the MVFF produces a state estimate by multiplying the filter gain $L$ and the augmented measurement vector $Z_N$. $L$ can be obtained using (26)–(31), where $F$ and $G$ are defined in (5); $H$ is defined in (6); and $Q(k)$ and $R(k)$ are process and measurement noise covariances, respectively. In addition, $Z_N$ can be obtained using (33). Combining measurements from $k = 1$ to $k = 4$, a large number of $Z_N$ can be obtained. The number of combinations for making $Z_N$ is $\prod_{i=1}^{4} m(k)$, which is the same as preliminary tracks produced by the MVFF. This method generates a significantly higher number of preliminary tracks than the MMTF. To effectively reduce the number of tentative tracks, we proposed a drastic method that selects one track for each measurement at a certain time. For a measurement $z_i(k)$, the most reliable track was selected by evaluation based on the Mahalanobis distance [21,22,25] defined as follows:

$$D_{i,th}(k) = [z_i(k) - \hat{z}_h(k)]^T R_i^{-1}(k) [z_i(k) - \hat{z}_h(k)],$$

$$\hat{z}_h(k) = H\hat{x}_h(k),$$

where $D_{i,th}(k)$ is the Mahalanobis distance [25] between the actual measurement $z_i(k)$ and the predicted measurement $\hat{z}_h(k)$. $x_h(k)$ is the estimate obtained from the $h$-th tentative track, where $h = 1, 2, \cdots, n(k)$, and $n(k)$ is the number of tentative tracks at time $k$. The estimate and covariance at time $k = 5$ are obtained using the MVFF and the combination of measurements (preliminary tracks) from $k = 1$ to $k = 4$, and those of the times from $k = 6$ to $k = N_W$ are obtained using the PDAF (see Appendix A). The estimate $\hat{x}_h(k)$ for
were merged by averaging to obtain the ultimate estimate and covariance. The overall with high likelihoods increases over time, which is an example of the survival of the fittest. A track in our method corresponds to the sample in particle filtering. Tracks are selected at each times step in the preliminary tracking (from $k = 6$ to $k = N_W$), in which the Mahalanobis distance is used as an evaluation tool. Each measurement has the authority to select a track that has the shortest Mahalanobis distance. A track may be selected for two

$$D_{lh}(k) = \left[ z_i(k) - \hat{x}_i^h(k-1) \right]^T R_i^{-1}(k) \left[ z_i(k) - \hat{x}_i^h(k-1) \right]$$

There are some similarities between the proposed track selection method and the resampling process in particle filtering. In the resampling process, a certain number of samples (i.e., state estimates) are selected based on the likelihood. In this process, a sample with a high likelihood may be repeatedly selected. Consequently, the number of samples with high likelihoods increases over time, which is an example of the survival of the fittest. The track in our method corresponds to the sample in particle filtering. Tracks are selected at each times step in the preliminary tracking (from $k = 6$ to $k = N_W$), in which the Mahalanobis distance is used as an evaluation tool. Each measurement has the authority to select a track that has the shortest Mahalanobis distance. A track may be selected for two or more measurements. Consequently, more favored (reliable) tracks can generate their duplicates, which can be viewed as the survival of the fittest.

At the initial time of preliminary tracking, the number of tentative tracks, $n(k)$, is significantly higher than the number of measurements, $m(k)$, in highly-heavy clutter environments. However, at $k = 5$, $n(k)$ drastically decreased to $m(k)$ by the selection based on the Mahalanobis distance. Moreover, $n(k)$ maintained the same value as $m(k)$ during the preliminary tracking. At the final time of the preliminary tracking, the tentative tracks were merged by averaging to obtain the ultimate estimate and covariance. The overall algorithm of the FFTF is summarized in Algorithm 1, where the a posteriori estimate and covariance were simply denoted as $\hat{x}(k)$ and $P(k)$, respectively.

**Algorithm 1: FFTF**

```
data: $Z(k) (k = 1, 2, \cdots, N_W)$
result: $\hat{x}^*(N_W), P^*(N_W)$
begin
  - Generate preliminary tracks (sequence of measurements) by combining measurements from $k = 1$ to $k = 4$.
  - Obtain the state estimate $\hat{x}(5)$ and covariance $P(5)$ using the MVFF (24)–(34) and the preliminary tracks.
  for $k = 6, 7, \cdots, N_W$ do
    for $i = 1, 2, \cdots, m(k)$ do
      - Obtain $R_i(k)$ using (4).
      for $h = 1, 2, \cdots, n(k)$ do
        - Compute the Mahalanobis distance for $\hat{x}_i^h(k-1)$:
          $\hat{x}_i^h(k-1) = H \hat{x}_i^h(k-1)$
          $D_{lh}(k-1) = \left[ z_i(k) - \hat{x}_i^h(k-1) \right]^T R_i^{-1}(k) \left[ z_i(k) - \hat{x}_i^h(k-1) \right]$
        end for
      - Select the tentative track that produced the minimal $D_{lh}(k-1)$ as the most reliable track for $z_i(k)$ ($\hat{x}_i(k-1)$ and $P_i(k-1)$).
    - Through filtering using the PDAF, obtain the state estimate $\hat{x}_i(k)$ and the covariance $P_i(k)$ using $\hat{x}_i(k-1)$, $P_i(k-1)$, and $Z(k)$.
  end for
  - Obtain the ultimate estimate $\hat{x}^*(N_W)$ and covariance $P^*(N_W)$:
    $\hat{x}^*(N_W) = \frac{1}{m(N_W)} \sum_{i=1}^{m(N_W)} \hat{x}_i(N_W)$.
    $P^*(N_W) = \frac{1}{m(N_W)} \sum_{i=1}^{m(N_W)} P_i(N_W)$.
end```
5. Simulation Results

In this section, the simulation results of preceding vehicle tracking using an automotive radar are presented to demonstrate the performance of the proposed FFTF. For the simulation, a typical situation of preceding vehicle tracking using the adaptive cruise control system on the highway was considered. The simulation scenario is as follows. The preceding and host vehicles are in the same lane. Various conditions for the relative distance, $D_r$, and velocity, $V_r$, between them were considered. In addition, the automotive radar was assumed to be the Delphi Electronically Scanning Radar (ESR), which has two operation modes. In the long-range mode, the radar exhibited an FOV and maximum detection range of 20° and 174 m, respectively. In the mid-range mode, the radar exhibited an FOV and maximum detection range of 90° and 60 m, respectively. Next, highly-heavy clutter environments were assumed, and the clutter density was set as $\lambda = 1.0 \times 10^{-1}$. This clutter density enabled the generation of more than 30 false measurements (clutters) at each time step. False measurements were generated so that they are independently and uniformly distributed in a square centered at the correct measurement. The area of the square $V$ was computed as follows:

$$ V = N_F / \lambda, $$(37)

$$ N_F = [10V(k)\lambda + 1]^-,$$ (38)

$$ V(k) = \pi \gamma |S(k)|^{1/2}, $$ (39)

where $N_F$ is the number of false measurements; the notation $[\cdot]^-$ indicates “rounded down to the nearest integer”; and $S(k)$ was obtained using KF under the assumption of an ideal clean (no clutters) environment [12].

The sampling interval was set as $T = 0.1$ s. The target detection probability was set as $P_D = 0.9$. The automotive radar reported range/bearing measurements with noise covariances, $\sigma_r = 0.25$ m and $\sigma_{\theta_{r\theta}} = 1.5^\circ$ [26,27]. The gate probability and the gate threshold were set as $P_G = 0.99$ and $\gamma = 9.21$, respectively. Trajectories of preceding vehicle were generated using the DWNA model (5) with $\sigma_w = 0.08$.

Using the aforementioned scenario and parameters, the preceding vehicle tracking was simulated only up to the track formation stage (1 ≤ $k$ ≤ $N_W$). The track formation was conducted using MMTF and FFTF, and their accuracies were compared in terms of the root-mean-square position error (RMSPE) and the root-mean-square velocity error (RMSVE) at the final time of track formation, $k = N_W$. The RMSPE and RMSVE are defined as follows:

$$ \text{RMSPE}(k) = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \sqrt{(x(k) - \hat{x}_i(k))^2 + (y(k) - \hat{y}_i(k))^2}, $$ (40)

$$ \text{RMSVE}(k) = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \sqrt{(\dot{x}_i(k) - \hat{\dot{x}}_i(k))^2 + (\dot{y}_i(k) - \hat{\dot{y}}_i(k))^2}, $$ (41)

where $N_{MC}$ is the number of Monte Carlo (MC) runs; $(x(k), y(k))$ and $((\dot{x}_i(k), \dot{y}_i(k)))$ are the true 2D positions and velocities, respectively; and $(\hat{x}_i(k), \hat{y}_i(k))$ and $(\hat{\dot{x}}_i(k), \hat{\dot{y}}_i(k))$ are the estimated 2D positions and velocities obtained the in $i$-th MC run, respectively.

For the window length of track formation, $N_W$, various values from 6 to 20 were tested. For each value of $N_W$, 100 MC simulations were performed, and the RMSPE and RMSVE at the final time of track formation were computed. The relative distance and velocity between the preceding and host vehicles were denoted as $D_r$ and $V_r$.

Figure 2 shows the simulation results under the long-range mode at a relative distance of $D_r = 100$ m, during which the distance was maintained ($V_r = 0$ km/h). The proposed FFTF exhibited significantly smaller RMSPE and RMSVE values than the MMTF algorithm. Moreover, the FFTF exhibited a consistent performance, whereas the RMSVE values of the MMTF decreased with a decrease in $N_W$. 

...
Figure 3 shows the simulation results under the mid-range mode at a relative distance of $D_r = 50$ m, during which the distance was maintained ($V_r = 0$ km/h). The proposed FFTF outperformed the MMTF under this condition, but the performance gap was smaller than that of the long-range condition. The largest performance gap was observed at $N_W = 6$.

In this study, different scenarios, in which the relative distance changed due to the different velocities of the two vehicles, were considered. The velocity of the preceding and host vehicles along the road were denoted as $V_p$ and $V_h$, respectively. Accordingly, the relative velocity was defined as $V_r = V_p - V_h$. Initially, $V_r = -20$ km/h was used, during which the two vehicles moved closer to each other. The initial distance between the vehicles was set to 100 m. Figure 4 shows the simulation results obtained under this scenario. The FFTF outperformed the MMTF, and the performance gap between both algorithms in terms of the RMSVE decreased as $N_W$ increased. In addition, as the window size of the track formation algorithm increases, the time consumption before the actual tracking began increased. This indicates that a large $N_W$ is not suitable.

Next, simulation was conducted under a scenario where the vehicles were getting farther (Figure 5). Here, $V_r$ was set to 20 km/h, and the initial $D_r$ was set to 100 m. The simulation results obtained under the “getting farther” scenario were similar to obtained under the “getting closer” scenario. In addition, FFTF outperformed MMFT, but the performance gap in terms of the RMSVE value decreased as $N_W$ increased. Because the sampling interval was $T = 0.1$ s, the time consumption for the track formation at $N_W = 20$ was 2 s. On the highway, a fast reaction is required, indicating that a time delay of 2 s is not suitable. In this study, using the FFTF at $N_W = 6$, both a fast reaction with a short time delay and accurate track formation were achieved.

Therefore, simulations were performed under more diverse $D_r$ and $V_r$ conditions at $N_W = 6$. Tables 1 and 2 show the simulation results obtained under various relative distances, which are represented more intuitively in Figure 6. Table 3 shows the results obtained with various relative velocities. The RMSPE and RMSVE values increased as $D_r$ and $V_r$ increased. Compared to the MMTF algorithm, the proposed FFTF algorithm exhibited not only a higher accuracy but also more consistent performance despite the changes in the $D_r$ and $V_r$ conditions.

![Figure 2. Cont.](image-url)
Figure 2. Simulation results under the long-range mode \((D_r = 100 \text{ m})\): (a) root-mean-square position error (RMSPE), (b) root-mean-square velocity error (RMSVE).

Figure 3. Simulation results under the mid-range mode \((D_r = 50 \text{ m})\): (a) RMSPE; (b) RMSVE.
Figure 4. Simulation results under the getting closer scenario ($V_r = -20$ km/h): (a) RMSPE; (b) RMSVE.

Figure 5. Cont.
Figure 5. Simulation results under getting farther scenario ($V_r = +20$ km/h): (a) RMSPE; (b) RMSVE.

Figure 6. Simulation results under various relative distances: (a) RMSPE; (b) RMSVE.
| Table 1. Simulation results under various relative distances in the long-range mode. |
|---|---|---|---|---|---|---|
| Performance | 100 m | 125 m | 150 m |
| Metric | MMTF | FFTF | MMTF | FFTF | MMTF | FFTF |
| RMSPE (m) | 8.29 | 3.00 | 8.85 | 3.17 | 9.48 | 3.43 |
| RMSVE (m/s) | 18.77 | 2.98 | 20.13 | 2.86 | 22.39 | 2.76 |

| Table 2. Simulation results under various relative distances in the mid-range mode. |
|---|---|---|---|---|
| Performance | 20 m | 40 m | 60 m |
| Metric | MMTF | FFTF | MMTF | FFTF | MMTF | FFTF |
| RMSPE (m) | 2.38 | 1.50 | 5.28 | 1.68 | 6.69 | 2.14 |
| RMSVE (m/s) | 6.15 | 2.51 | 12.24 | 2.90 | 14.77 | 3.13 |

| Table 3. Simulation results under various relative velocities in the long-range mode. |
|---|---|---|---|---|---|---|
| Performance | −30 km/h | −10 km/h | +10 km/h | +30 km/h |
| Metric | MMTF | FFTF | MMTF | FFTF | MMTF | FFTF | MMTF | FFTF |
| RMSPE (m) | 8.10 | 3.09 | 7.85 | 3.04 | 8.55 | 3.15 | 8.65 | 3.01 |
| RMSVE (m/s) | 18.41 | 9.02 | 17.86 | 4.53 | 18.90 | 4.16 | 19.79 | 9.06 |

6. Conclusions

This study proposed a novel track formation algorithm, namely FFTF, for preceding vehicle tracking using automotive radars. In the proposed FFTF algorithm, first, the initial estimate and covariance were obtained using a type of FIR filter (i.e., MVFF). The FFTF effectively reduced the number of preliminary / tentative tracks using the Mahalanobis distance-based evaluation. In the simulations, the FFTF was significantly more accurate than the MMTF in terms of both RMSPE and RMSVE. The FFTF exhibited consistent performance, whereas the MMTF exhibited large changes in performance depending on the window size, relative distance and velocities. Therefore, the FFTF algorithm is expected to exhibit improved performance for preceding vehicle tracking in automotive radar systems. However, the FFTF requires more computation time for track initiation process compared to the MMTF. We intend to improve the FFTF by adopting a new FIR filter in future work.

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:

- FIR: Finite Impulse Response
- PDAF: Probabilistic Data Association Filter
- IMM: Interacting Multiple Model
- MMTF: Multiple Model Track Formation
- FFTF: FIR Filter-Based Track Formation
Appendix A. Probabilistic Data Association Filter

The probabilistic data association filter (PDAF) algorithm commonly used in the MMTF and FFTF is introduced in this appendix. The PDAF recursively performed the following processes. The state estimate and covariance were updated using the motion model:

$$\hat{x}(k|k-1) = F\hat{x}(k-1|k-1),$$
$$P(k|k-1) = FP(k-1|k-1)F^T + GQ(k)G^T. \quad (A1)$$

The measurements within the validation region \((i = 1, \cdots, m(k))\) are selected:

$$v_i(k) = (z_i(k) - \hat{z}(k))^T S_i^{-1}(k) (z_i(k) - \hat{z}(k)) \leq \gamma,$$
$$S_i(k) = HP(k|k-1)H^T + R_i(k), \quad (A2)$$

where \(v_i(k)\) is the innovation for \(z_i(k)\); \(\gamma\) is the gate threshold; and \(S_i(k)\) is the innovation covariance. The gate threshold \(\gamma\) can be obtained from a chi-square table.

The association probability for \(z_i(k)\) being the correct measurement is computed:

$$\beta_i(k) = \begin{cases} L_i(k) & i = 1, \cdots, m(k) \\ \frac{1 - P_D \sum_{j=1}^{m(k)} L_j(k)}{1 - P_D \sum_{j=1}^{m(k)} L_j(k)} & i = 0 \end{cases} \quad (A3)$$

where \(i = 0\) corresponds to “none is correct”, and \(m(k)\) is the number of measurements \([12, 28]\). The likelihood ratio \(L_i(k)\) is computed:

$$L_i(k) = \frac{\exp[-0.5(z_i(k) - \hat{z}(k))^T S_i^{-1}(k) (z_i(k) - \hat{z}(k))]}{(2\pi)|R_i(k)|^{1/2} m(k)} P_D V_i(k)$$
$$V_i(k) = c_{n, \gamma} |S_i(k)|^{1/2}, \quad (A4)$$

where \(V_i(k)\) is the volume of the validation region. The state estimate is updated using the measurements:

$$\hat{x}(k|k) = \hat{x}(k|k-1) + \sum_{i=1}^{m(k)} W_i(k) \beta_i(k) v_i(k),$$
$$W_i(k) = P(k|k-1)H^T S_i^{-1}(k),$$
$$v_i(k) = z_i(k) - \hat{z}(k). \quad (A5)$$
In addition, the estimation error covariance was updated as follows [28]:

$$P(k|k) = \bar{P}(k|k) + \hat{P}(k|k),$$

$$P(k|k) = \beta_0(k) P(k|k-1) + \sum_{i=1}^{m} \beta_i(k) (I - W_i(k)H) P(k|k-1),$$

$$\hat{P}(k|k) = \sum_{i=1}^{m} \beta_i(k) W_i(k)v_i(k)v_i^T(k)W_i^T(k)$$

$$- \left( \sum_{i=1}^{m} \beta_i(k) W_i(k)v_i(k) \right) \left( \sum_{i=1}^{m} \beta_i(k) v_i^T(k) W_i^T(k) \right).$$  (A6)

If there is no measurement in the validation region, the PDAF does not perform the measurement update processes.

References

1. Kim, W.; Cho, H.; Kim, J.; Kim, B.; Lee, S. YOLO-Based Simultaneous Target Detection and Classification in Automotive FMCW Radar Systems. Sensors 2020, 20, 2897. [CrossRef] [PubMed]
2. Hau, F.; Baumgärtner, F.; Vossiek, M. The Degradation of Automotive Radar Sensor Signals Caused by Vehicle Vibrations and Other Nonlinear Movements. Sensors 2020, 20, 6195. [CrossRef]
3. Wu, Q.; Gao, T.; Lai, Z.; Li, D. Hybrid SVM-CNN Classification Technique for Human-Vehicle Targets in an Automotive LFMCW Radar. Sensors 2020, 20, 3504. [CrossRef]
4. Kui, L.; Huang, S.; Feng, Z. Interference Analysis for mmWave Automotive Radar Considering Blockage Effects. Sensors 2021, 21, 3962. [CrossRef] [PubMed]
5. Lee, S.; Lee, J.Y.; Kim, S.C. Mutual Interference Suppression Using Wavelet Denoising in Automotive FMCW Radar Systems. IEEE Trans. Ind. Electron. 2021, 68, 887–897. [CrossRef]
6. Waldschmidt, C.; Hasch, J.; Menzel, W. Automotive Radar—From First Efforts to Future Systems. IEEE J. Microwaves 2021, 1, 135–148. [CrossRef]
7. Lee, S.; Lee, B.H.; Lee, J.E.; Kim, S.C. Statistical Characteristic-Based Road Structure Recognition in Automotive FMCW Radar Systems. IEEE Trans. Ind. Transp. Syst. 2019, 20, 2418–2429. [CrossRef]
8. Patole, S.M.; Torlak, M.; Wang, D.; Ali, M. Automotive radars: A review of signal processing techniques. IEEE Signal Process. Mag. 2017, 34, 22–35. [CrossRef]
9. Chen, Y.; Wang, Y.; Qu, F.; Li, W. A Graph-Based Track-Before-Detect Algorithm for Automotive Radar Target Detection. IEEE Sens. J. 2021, 21, 6587–6599. [CrossRef]
10. Jung, J.; Lee, S.; Lim, S.; Kim, S.C. Machine Learning-Based Estimation for Tilted Mounting Angle of Automotive Radar Sensor. IEEE Sens. J. 2020, 20, 2928–2937. [CrossRef]
11. Bar-Shalom, Y.; Tse, E. Tracking in a Cluttered Environment with Probabilistic Data Association. Automatica 1975, 11, 451–460. [CrossRef]
12. Bar-Shalom, Y.; Willett, P.K.; Tian, X. Tracking and Data Fusion—A Handbook of Algorithms; YBS Publishing: Storrs, CT, USA, 2011.
13. Fortmann, 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]
14. Blackman, S.S. Multiple Hypothesis Tracking for Multiple Target Tracking. IEEE Trans. Aerosp. Electron. Syst. 2004, 19, 5–18. [CrossRef]
15. Mahler, R.P.S. Statistical Multisource-Multitarget Information Fusion; Artech House: Norwood, MA, USA, 2007.
16. 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]
17. 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]
18. 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]
19. Pak, J.M.; Ahn, C.K. Dual-Mode Deadbeat H2 FIR Filtering for Discrete-Time Systems. Signal Process. 2021, 184, 108057. [CrossRef]
20. Pak, J.M. Hybrid PDA/FIR Filtering for Preceding Vehicle Tracking Using Automotive Radars. IEEE Access 2021, 9, 118726–118735. [CrossRef]
21. 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]
22. 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]
23. 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]

24. 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]

25. Mahalanobis, P.C. On the generalised distance in statistics. *Proc. Natl. Inst. Sci. India* 1936, 2, 49–55.

26. 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]

27. 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]

28. Mcgee, J.A.; Lugnbuhl, T.E.; Dibiase, J.H.; Ainsleigh, P.L. A Modified PDAF Algorithm For Measurement Covariances that Vary within a Time Scan. *IEEE Trans. Aerosp. Electron. Syst.* 2012, 48, 906–912. [CrossRef]