

IMM-based Multiple Object Tracking using a State Prediction Neural Network

^{1st} Chan-Bin Lim

*Robotics Program
Korea Advanced Institute of
Science and Technology (KAIST)
Daejeon, Republic of Korea, 34051*

^{2nd} Dong-Hee Paek

*Mechanical Engineering Research
Institute
Korea Advanced Institute of
Science and Technology (KAIST)
Daejeon, Republic of Korea, 34141*

^{3rd} Seung-Hyun Kong*

*Cho Chun Shik Graduate School of
Mobility
Korea Advanced Institute of
Science and Technology (KAIST)
Daejeon, Republic of Korea, 34051*

Abstract—Object tracking is essential for autonomous vehicles to avoid obstacles and plan routes. Radar maintains detection performance even in adverse weather and can measure relative velocity through the Doppler effect, making it well suited for object tracking. In this paper, we propose a data-driven state PRedictor-based Interacting Multiple Model tracking method (PR-IMM) that improves nonlinear object-motion representation while preserving the stability and interpretability of physics-based motion models. The proposed method employs a Transformer-based PRediction model (PR) that incorporates radar Doppler measurements to predict object displacement. The PR model is integrated into the IMM as a mode alongside the CV, CA, and CT motion models, and their prior positions are dynamically combined according to the mode probabilities. Experimental results show that PR-IMM reduces position-estimation error by 57.3% over the IMM and by 16.5% over the PR, while reducing ID switches by 25.3% and improving IDF1 by 9.6% over the IMM.

Index Terms—Radar, multiple object tracking, Kalman filter, interacting multiple model, Transformer.

I. INTRODUCTION

Object tracking detects objects from sensor data and continuously estimates their locations. With the development of autonomous driving and advanced driver assistance systems (ADAS), it has become a core technology for safe driving decisions [1]–[3]. Cameras, LiDAR, radar, and other sensors are widely used for this purpose. Among them, radar has attracted particular attention for object tracking because it maintains robust detection performance in adverse weather such as rain and fog and can directly measure relative velocity through the Doppler effect [4]–[6].

Accordingly, various approaches have been proposed for radar-based object tracking, and early studies mainly relied on motion models implemented with Kalman filters (KF) [7], [8]. Representative models include Constant Velocity (CV), Constant Acceleration (CA), and Constant Turn Rate (CT). Each assumes a specific motion pattern and defines a state-transition model to predict the next object position. Because a single motion model cannot flexibly represent diverse motion patterns, the Interacting Multiple Model (IMM), which dynamically combines multiple motion models according to their mode probabilities, has been widely adopted [9]. However, motion-model-based approaches depend on predefined motion patterns and therefore have limited ability to represent the nonlinear object motions that occur in real road environments [10].

To overcome these limitations, recent studies have actively applied deep-learning-based trajectory prediction models to object tracking [11]. In particular, Transformer-based models [12] effectively model long-range dependencies in time-series data through self-attention and have shown strong performance in predicting future positions from past trajectories [13]. Because such data-driven models do not assume a particular motion pattern, they can learn nonlinear motions that are difficult to describe with CV, CA, or CT models. On the other hand, their predictions can become unstable when observation noise is large or when the input lies outside the training distribution [14].

Motion-model-based and data-driven approaches therefore have complementary strengths. Data-driven models can directly learn diverse motion patterns without predefining dynamic assumptions, while the Kalman filters underlying physics-based motion models provide stable state estimation by explicitly handling measurement uncertainty through covariance. Combining these strengths is thus a promising way to improve object-tracking performance.

From this perspective, we propose a PR (PRediction model)-IMM tracker that integrates a Transformer-based prediction model into the IMM framework. The proposed method uses a data-driven prior state from a Transformer predictor learned from radar measurements together with physics-based prior states from CV, CA, and CT models. These priors are probabilistically combined through IMM mode probabilities so that the tracker can adaptively exploit the most suitable information under each motion condition, targeting robust tracking performance that cannot be achieved by a single motion model or a data-driven predictor alone.

The main contributions of this paper are as follows:

- We propose a Transformer-based network that considers radar Doppler measurements for accurate state prediction.
- We propose a PR-IMM multiple-object-tracking framework in which a data-driven Transformer predictor and physics-based CV, CA, and CT motion models operate as IMM modes.
- Using the large-scale radar point-cloud dataset RadarScenes [15], we demonstrate that the proposed method significantly improves both position-estimation accuracy and object-tracking performance over single models and a conventional IMM (CV+CA+CT).

The remainder of this paper is organized as follows. Section II reviews related work. Section III describes the proposed PR-

*Corresponding author: skong@kaist.ac.kr

IMM tracker in detail. Section IV presents the experimental setup and results, and Section V concludes the paper.

II. RELATED WORKS

This section reviews representative object-tracking methods based on Kalman filters, IMM, deep learning, and combinations of Kalman filtering and neural networks.

A. Kalman-Filter-Based Object Tracking

CV, CA, and CT motion models are among the most common choices in Kalman-filter-based tracking. SORT combines a CV-model Kalman filter with the Hungarian algorithm for real-time multiple-object tracking^[16], while CA-model Kalman filtering has also been used to estimate vehicle motion states from automotive radar measurements^[17]. The CV model is robust for constant-speed motion but weak under abrupt acceleration. The CA model mitigates this limitation but can accumulate larger inertial errors in the lateral direction. The CT model is advantageous for turning maneuvers but may exhibit larger errors during straight constant-speed motion. Thus, a single motion model is optimized for a particular motion assumption and inevitably involves a trade-off; prediction errors can increase in real traffic where objects combine straight motion, acceleration, deceleration, and turning.

B. IMM-Based Object Tracking

IMM-based tracking commonly combines CV, CA, and CT models. Previous studies have applied IMM to 3D multiple-object tracking to reduce ID switches^[18] and to improve tracking under diverse motion patterns^[19]. By dynamically weighting several motion models according to their mode probabilities, IMM can handle a broader range of motions than a single model. Nevertheless, when an object performs a maneuver outside the scope of all predefined models, the likelihoods of all modes decrease. The subsequent weighted combination is then formed from models that each approximate the true motion poorly, resulting in increased state-estimation error.

C. Deep-Learning-Based Object Tracking

Deep-learning-based object-tracking methods frequently predict the next position with a neural network and then use the prediction for data association^{[20], [21]}. Recurrent networks such as LSTMs were initially used as predictors that replace the motion model of a Kalman filter^[22]. More recently, Transformer-based predictors have attracted attention because self-attention can capture longer-range temporal dependencies more effectively than recurrent models^[13]. However, these predictors output a future position without explicitly providing how reliable that prediction is. Consequently, when a prediction deviates substantially from the actual observation, no intrinsic uncertainty-based mechanism is available to correct it, and the error can accumulate throughout tracking.

D. Existing Combinations of Kalman Filtering and Deep Learning

Recent work has also combined Kalman filtering with deep learning. Representative examples include a neural network coupled to the Kalman-filter state model^[23] and an LSTM-based error-compensation method for Kalman filtering^[24]. In these approaches, however, deep learning is used primarily as an auxiliary component of the Kalman filter. When the

learned prediction becomes inaccurate, there is no independent alternative mode that can replace or compensate for it, which can degrade tracking performance.

III. PR-IMM-BASED MULTIPLE OBJECT TRACKING

This section presents the proposed PR-IMM tracker, which integrates a Transformer-based prediction model and Kalman-filter-based motion models into a single IMM framework. The Transformer predictor learns nonlinear object motion from data, while the CV, CA, and CT models provide physics-based motion priors that can compensate when the learned prediction becomes unreliable. The goal is to achieve robust position estimation even under noisy observations.

A. Overall System Pipeline

The proposed object-tracking pipeline is shown in Fig. 1. We assume that object detection has already been completed^{[25], [26]} and focus on the subsequent tracking stage. Instead of running a detector, we use the point-wise class IDs and track IDs provided by RadarScenes^[15]. Points with the same track ID are grouped into one object point cloud, and the point values are averaged to obtain an object-level radar measurement vector. At time step k , this vector is defined as

$$\mathbf{m}_k = [x_k, y_k, v_{r,k}, \text{RCS}_k]^T, \quad (1)$$

where x_k and y_k denote the longitudinal and lateral distances of the object, respectively, $v_{r,k}$ is radial velocity in the radar coordinate system, and RCS_k is the Radar Cross Section.

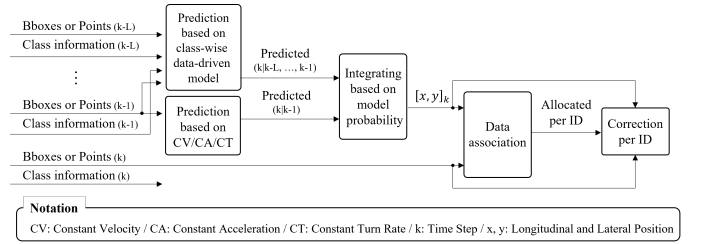


Fig. 1. Overall framework of PR-IMM.

B. Kalman-Filter Prediction with Motion Models

This subsection describes the Kalman-filter prediction stage based on the CV, CA, and CT motion models. All three models use the two-dimensional object position as the Kalman-filter observation, with

$$\mathbf{z}_k = [x_k, y_k]^T. \quad (2)$$

Each motion model $i \in \{\text{CV}, \text{CA}, \text{CT}\}$ uses its own state vector and transition function. The Kalman prediction consists of state prediction

$$\mathbf{x}_{i,k}^- = \mathbf{f}_i(\mathbf{x}_{i,k-1}), \quad (3)$$

and covariance prediction

$$\mathbf{P}_{i,k}^- = \mathbf{F}_{i,k} \mathbf{P}_{i,k-1} \mathbf{F}_{i,k}^T + \mathbf{Q}_i, \quad (4)$$

where $\mathbf{x}_{i,k}^-$ is the prior state predicted by model i from the previous posterior state $\mathbf{x}_{i,k-1}$. The covariance $\mathbf{P}_{i,k}^-$ represents the uncertainty of the prior state after incorporating the predefined process-noise covariance $\mathbf{Q}_i$. The superscript $(\cdot)^-$ denotes a value before the Kalman correction. For CV and

CA, $f_i(\mathbf{x}) = \mathbf{F}_i \mathbf{x}$ is linear, whereas CT uses the nonlinear transition in Eqs. (10)–(13) and propagates covariance using its Jacobian.

1) *CV Model*: The CV model assumes straight-line motion at constant velocity. Its state vector contains longitudinal and lateral positions and velocities:

$$\mathbf{x}_{CV,k}^- = [x_k^-, v_{x,k}^-, y_k^-, v_{y,k}^-]^T. \quad (5)$$

The initial velocity components v_x and v_y are initialized from the measured radar radial velocity v_r and subsequently refined by Kalman correction. Using the time interval Δt_k , the state-transition matrix is

$$\mathbf{F}_{CV,k} = \begin{bmatrix} 1 & \Delta t_k & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t_k \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (6)$$

2) *CA Model*: The CA model assumes constant acceleration and augments the CV state with acceleration components:

$$\mathbf{x}_{CA,k}^- = [x_k^-, v_{x,k}^-, a_{x,k}^-, y_k^-, v_{y,k}^-, a_{y,k}^-]^T. \quad (7)$$

The initial accelerations are set to zero and are subsequently corrected using measurements. The corresponding transition matrix is

$$\mathbf{F}_{CA,k} = \begin{bmatrix} 1 & \Delta t_k & \frac{1}{2} \Delta t_k^2 & 0 & 0 & 0 \\ 0 & 1 & \Delta t_k & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t_k & \frac{1}{2} \Delta t_k^2 \\ 0 & 0 & 0 & 0 & 1 & \Delta t_k \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (8)$$

3) *CT Model*: The CT model assumes a constant turn rate ω and uses

$$\mathbf{x}_{CT,k}^- = [x_k^-, v_{x,k}^-, y_k^-, v_{y,k}^-, \omega_k^-]^T. \quad (9)$$

The initial turn rate is set to zero and is later refined through Kalman correction. Defining $\phi_k = \omega_{k-1} \Delta t_k$, the nonlinear transition for $\omega_{k-1} \neq 0$ is

$$x_k^- = x_{k-1} + \frac{\sin \phi_k}{\omega_{k-1}} v_{x,k-1} - \frac{1 - \cos \phi_k}{\omega_{k-1}} v_{y,k-1}, \quad (10)$$

$$v_{x,k}^- = \cos \phi_k v_{x,k-1} - \sin \phi_k v_{y,k-1}, \quad (11)$$

$$y_k^- = y_{k-1} + \frac{1 - \cos \phi_k}{\omega_{k-1}} v_{x,k-1} + \frac{\sin \phi_k}{\omega_{k-1}} v_{y,k-1}, \quad (12)$$

$$v_{y,k}^- = \sin \phi_k v_{x,k-1} + \cos \phi_k v_{y,k-1}. \quad (13)$$

When ω_{k-1} is close to zero, a Taylor-series approximation is used to avoid numerical divergence, and covariance propagation is performed using the Jacobian of Eqs. (10)–(13).

C. Transformer-Based Prediction

This subsection describes the Transformer-based prediction model, denoted PR, which predicts the displacement of an object at the next time step from radar observations. The input feature at historical index l is

$$\mathbf{s}_l = [x_l - x_{k-1}, y_l - y_{k-1}, v_{r,l}, \text{RCS}_l, t_l - t_k]^T, \quad (14)$$

where $l = k-L, \dots, k-1$. The PR model uses the previous L radar observations before prediction time k , with a maximum sequence length of $L = 5$. Each $\mathbf{s}_l$ consists of the relative position with respect to the most recent observation at $k-1$, radial velocity, RCS, and relative time to prediction time k .

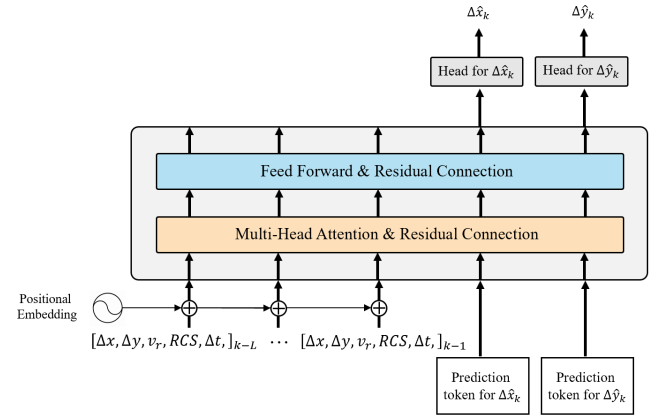


Fig. 2. Architecture of the PR model.

1) *PR Model Architecture*: Figure 2 shows the proposed PR architecture. To make the model learn motion patterns rather than absolute scene coordinates, all historical positions are represented relative to the latest observation $\mathbf{z}_{k-1}$. Radial velocity and RCS provide motion and reflection characteristics of the object, and the actual time interval is used to estimate displacement precisely.

Because a Transformer processes an input sequence in parallel, temporal order is not explicitly encoded. Positional embeddings are therefore added to each historical input so that the network receives the temporal ordering of the past trajectory. Two learnable query tokens, q_x and q_y , are also appended to the input to predict the x - and y -direction displacements at the next time step.

The input sequence and query tokens pass through a Transformer encoder composed of Multi-Head Attention and Feed-Forward layers with residual connections. From the final outputs corresponding to the two query tokens, the model predicts $\Delta \hat{\mathbf{p}}_{PR,k} = [\Delta \hat{x}_k, \Delta \hat{y}_k]^T$.

2) *PR Filter*: The PR prior position at time k is obtained by adding the predicted displacement to the posterior PR position at time $k-1$:

$$\mathbf{p}_{PR,k}^- = \mathbf{p}_{PR,k-1} + \Delta \hat{\mathbf{p}}_{PR,k}. \quad (15)$$

As in the CV, CA, and CT modes, the covariance of the PR prior state represents prediction uncertainty and is defined as

$$\mathbf{P}_{PR,k}^- = \mathbf{P}_{PR,k-1} + \mathbf{Q}_{PR}. \quad (16)$$

D. Prior State Fusion

This subsection explains how the motion-model predictions in Section III-B and the PR prediction in Section III-C are combined. PR-IMM contains four modes: PR, CV, CA, and CT. Although their complete state vectors have different dimensions, the position component of each prior state is projected through the observation matrix $\mathbf{H}_i$ and used for state fusion: $\mathbf{p}_{i,k}^- = \mathbf{H}_i \mathbf{x}_{i,k}^- = [x_{i,k}^-, y_{i,k}^-]^T$.

1) *Input Mixing*: Before the PR, CV, CA, and CT modes predict the next state, each performs input mixing using the previous mode probabilities and mode-transition probabilities. PR-IMM maintains the modes independently, but an object's motion pattern can change over time. To preserve continuity when the dominant mode changes, the posterior positions and position covariances of all modes at the previous time step are combined and used as the initial values for each destination mode. Thus, mode j begins prediction from the mixed position $\bar{\mathbf{p}}_{j,k-1}$, which reflects not only its own previous state but also the posterior states of other modes. The overbar denotes a mode-weighted mean.

Let π_{ij} denote the transition probability from mode i to mode j , and let $\mu_{i,k-1}$ be the posterior mode probability at the previous time step. The prior probability that mode j is selected at the current time step is

$$c_{j,k} = \sum_{r=1}^M \pi_{rj} \mu_{r,k-1}, \quad (17)$$

where M is the total number of modes, namely PR, CV, CA, and CT. The mixing probability that quantifies the contribution of previous mode i to the initial state of current mode j is

$$\beta_{ij,k-1} = \frac{\pi_{ij} \mu_{i,k-1}}{c_{j,k}}. \quad (18)$$

Using Eq. (18), the mixed initial position and position covariance of mode j are

$$\bar{\mathbf{p}}_{j,k-1} = \sum_{i=1}^M \beta_{ij,k-1} \mathbf{p}_{i,k-1}, \quad (19)$$

$$\begin{aligned} \bar{\mathbf{P}}_{j,k-1}^p &= \sum_{i=1}^M \beta_{ij,k-1} \left[\mathbf{P}_{i,k-1}^p + (\mathbf{p}_{i,k-1} - \bar{\mathbf{p}}_{j,k-1}) \right. \\ &\quad \left. \times (\mathbf{p}_{i,k-1} - \bar{\mathbf{p}}_{j,k-1})^T \right]. \end{aligned} \quad (20)$$

Equation (20) includes the dispersion caused by disagreement among the mode positions, so the initial position covariance becomes more conservative when the modes disagree strongly. Because the complete states of the modes have different dimensions and structures, input mixing is performed only for the common position and position covariance.

Each mode independently preserves its non-position state variables. The mixed full state is therefore written as

$$\bar{\mathbf{x}}_{j,k-1} = [\bar{\mathbf{p}}_{j,k-1}, \mathbf{h}_{j,k-1}]^T, \quad (21)$$

where $\mathbf{h}_{j,k-1}$ contains the non-position states of destination mode j at time $k-1$.

To maintain consistency between the mixed position covariance and the full covariance, the cross-covariance between position and the remaining states and the covariance of the

remaining states are updated using the conditional relationship of the original covariance. Before input mixing, the full covariance is partitioned as

$$\mathbf{P}_{j,k-1} = \begin{bmatrix} \mathbf{P}_{pp,j,k-1} & \mathbf{P}_{ph,j,k-1} \\ \mathbf{P}_{hp,j,k-1} & \mathbf{P}_{hh,j,k-1} \end{bmatrix}, \quad (22)$$

where $\mathbf{P}_{pp}$ is the position covariance, $\mathbf{P}_{ph} = \mathbf{P}_{hp}^T$ is the position/non-position cross-covariance, and $\mathbf{P}_{hh}$ is the covariance of the remaining states. We define

$$\mathbf{B}_{j,k-1} = \mathbf{P}_{hp,j,k-1} (\mathbf{P}_{pp,j,k-1})^\dagger, \quad (23)$$

$$\mathbf{C}_{j,k-1} = \mathbf{P}_{hh,j,k-1} - \mathbf{B}_{j,k-1} \mathbf{P}_{pp,j,k-1} \mathbf{B}_{j,k-1}^T, \quad (24)$$

and reconstruct the mixed full covariance as

$$\bar{\mathbf{P}}_{j,k-1} = \begin{bmatrix} \bar{\mathbf{P}}_{j,k-1}^p & \bar{\mathbf{P}}_{j,k-1}^p \mathbf{B}_{j,k-1}^T \\ \mathbf{B}_{j,k-1} \bar{\mathbf{P}}_{j,k-1}^p & \mathbf{C}_{j,k-1} + \mathbf{B}_{j,k-1} \bar{\mathbf{P}}_{j,k-1}^p \mathbf{B}_{j,k-1}^T \end{bmatrix}. \quad (25)$$

Here, $(\cdot)^\dagger$ denotes the pseudoinverse.

2) *PR-IMM Prior State Fusion*: Starting from the mixed initial values in Eqs. (21) and (25), each mode independently predicts its prior state. The PR-IMM prior position at time k is then obtained by weighting the predicted position of each mode by its current prior mode probability:

$$\mathbf{p}_{\text{PR-IMM},k}^- = \sum_{i=1}^M c_{i,k} \mathbf{p}_{i,k}^-. \quad (26)$$

The fused prior position in Eq. (26) is used to match tracks and observations in the data-association stage.

E. Data Association

This subsection describes how the PR-IMM prior state is used to associate existing tracks with new observations. Radar-based multiple-object tracking receives multiple observations at each time step, making correct observation-to-track assignment essential. For track a and observation j , we define an association cost using the Euclidean distance between the PR-IMM prior position $\mathbf{p}_{\text{PR-IMM},a,k}^-$ and observation position $\mathbf{z}_{j,k} = [x_{j,k}, y_{j,k}]^T$ together with their radial-velocity difference:

$$\begin{aligned} d_{a,j,k} &= \left\| \mathbf{z}_{j,k} - \mathbf{p}_{\text{PR-IMM},a,k}^- \right\|_2 \\ &\quad + \lambda_v \min(|v_{r,j,k} - v_{r,a}^{\text{last}}|, v_{\max}). \end{aligned} \quad (27)$$

Here, $v_{r,a}^{\text{last}}$ is the radial velocity of the radar measurement most recently associated with track a . This term helps distinguish spatially close objects moving at different velocities. To prevent velocity from dominating the total association cost, the velocity difference is capped by $v_{\max}$ and weighted by λ_v . We use $\lambda_v = 0.1$ and $v_{\max} = 5$ m/s. Because RCS varies substantially with object pose, viewing angle, and material composition, it is used only as an input feature to the PR model and is not included in the data-association cost.

For each observation j , Eq. (27) is used to construct a cost matrix between tracks and observations belonging to the same object class. The Hungarian algorithm then performs a global one-to-one assignment. Associated tracks proceed to the posterior-state-fusion stage using the assigned observation,

while observations not associated with any existing track initialize new tracks.

F. Posterior State Fusion

This subsection explains how each mode is corrected after data association, how the posterior mode probabilities are updated, and how the final position is obtained.

1) *Mode Probability Update*: Once an observation is assigned to a track, each mode computes the innovation $\nu_{i,k}$ between its prior prediction and the observation and the corresponding innovation covariance $\mathbf{S}_{i,k}$. These quantities are used both for Kalman correction and for computing the mode likelihood and posterior mode probability. For PR, CV, CA, and CT,

$$\nu_{i,k} = \mathbf{z}_k - \mathbf{P}_{i,k}^-, \quad (28)$$

$$\mathbf{S}_{i,k} = \mathbf{H}_i \mathbf{P}_{i,k}^- \mathbf{H}_i^T + \mathbf{R}_i, \quad (29)$$

where $\mathbf{H}_i$ maps the prior state of mode i to the position observation and $\mathbf{R}_i$ is the radar measurement-noise covariance. For the PR mode, $\mathbf{Q}_{\text{PR}}$ is set smaller than $\mathbf{R}_{\text{PR}}$ so that the learned prior state can be treated as relatively reliable compared with the raw observation.

The Gaussian likelihood measuring how well mode i explains the current observation is

$$\Lambda_{i,k} = \frac{1}{\sqrt{(2\pi)^2 |\mathbf{S}_{i,k}|}} \exp\left(-\frac{1}{2} \nu_{i,k}^T \mathbf{S}_{i,k}^{-1} \nu_{i,k}\right). \quad (30)$$

Using this likelihood, the mode probability is updated by

$$\mu_{i,k} = \frac{\Lambda_{i,k}^\tau c_{i,k}}{\sum_{j=1}^M \Lambda_{j,k}^\tau c_{j,k}}, \quad (31)$$

where τ controls the influence of the likelihood. We use $\tau = 0.1$.

2) *Correction*: Using the PR-IMM prior state and the associated observation $\mathbf{z}_k$, each mode updates its posterior state $\mathbf{x}_{i,k}$ and covariance $\mathbf{P}_{i,k}$. The Kalman gain is

$$\mathbf{K}_{i,k} = \mathbf{P}_{i,k}^- \mathbf{H}_i^T \mathbf{S}_{i,k}^{-1}, \quad (32)$$

followed by the posterior-state update

$$\mathbf{x}_{i,k} = \mathbf{x}_{i,k}^- + \mathbf{K}_{i,k} \nu_{i,k}, \quad (33)$$

and the Joseph-form posterior-covariance update

$$\mathbf{P}_{i,k} = (\mathbf{I} - \mathbf{K}_{i,k} \mathbf{H}_i) \mathbf{P}_{i,k}^- (\mathbf{I} - \mathbf{K}_{i,k} \mathbf{H}_i)^T + \mathbf{K}_{i,k} \mathbf{R}_i \mathbf{K}_{i,k}^T. \quad (34)$$

Although the observation vector contains only position, the Kalman gain in Eq. (32) is computed from each mode's full prior covariance. Therefore, the position innovation also updates velocity, acceleration, and turn-rate states through their cross-covariances with position.

3) *PR-IMM Posterior State Fusion*: The final PR-IMM position is obtained by weighting the posterior position of every mode by the posterior mode probability from Eq. (31):

$$\mathbf{p}_{\text{PR-IMM},k} = \sum_{i=1}^M \mu_{i,k} \mathbf{H}_i \mathbf{x}_{i,k}. \quad (35)$$

Thus, the PR, CV, CA, and CT position estimates are dynamically combined according to the motion situation at each time step.

IV. EXPERIMENTS

This section quantitatively and qualitatively evaluates the proposed PR-IMM tracker and analyzes the contribution of its components.

A. Experimental Setup

1) *Dataset*: We train the PR model and evaluate the tracker on RadarScenes^[15], a large-scale radar point-cloud dataset collected in autonomous-driving environments. RadarScenes provides point-wise segmentation labels and track IDs for five object classes: Car, Large Car, Pedestrian, Pedestrian Group, and Two Wheeler. We merge Car and Large Car into a Car category and Pedestrian and Pedestrian Group into a Pedestrian category, and evaluate only these Car and Pedestrian categories.

RadarScenes contains 158 sequences and is split by sequence for training and evaluation. The validation set uses sequences 6, 42, 58, 85, 99, and 122. The evaluation set uses sequences 1, 2, 3, 4, 16, 33, 38, 41, 47, 49, 57, 71, 81, 98, 105, 112, 113, 120, 123, 125, 131, and 146. The remaining 130 sequences form the training set. Table I summarizes the split and class composition.

TABLE I
DATASET SPLIT AND CLASS COMPOSITION.

Split	Sequences	Frames	Car object frames	Pedestrian object frames
Train	130	32,555	49,349	60,216
Validation	6	1,291	1,591	1,744
Evaluation	22	4,784	11,307	6,559

2) *Compared Methods*: To evaluate PR-IMM, we quantitatively compare six methods in total. The single-motion-model baselines are CV, CA, and CT Kalman filters. The conventional IMM (CV+CA+CT) serves as the multi-model baseline. We also include the standalone Transformer predictor PR to contrast the limitations of purely physics-based and purely data-driven approaches with the proposed hybrid framework.

3) *Evaluation Metrics*: Position-prediction performance is evaluated using Root Mean Squared Error (RMSE), while identity preservation in multiple-object tracking is evaluated using ID switches (IDSW) and IDF1.

The longitudinal error RMSE_x , lateral error RMSE_y , and planar distance error RMSE_{xy} are

$$\text{RMSE}_x = \sqrt{\frac{1}{N} \sum_{k=1}^N (x_k - \hat{x}_k)^2}, \quad (36)$$

$$\text{RMSE}_y = \sqrt{\frac{1}{N} \sum_{k=1}^N (y_k - \hat{y}_k)^2}, \quad (37)$$

$$\text{RMSE}_{xy} = \sqrt{\frac{1}{N} \sum_{k=1}^N [(x_k - \hat{x}_k)^2 + (y_k - \hat{y}_k)^2]}, \quad (38)$$

where N is the total number of samples.

IDSW is incremented by one when the estimated track ID associated with a ground-truth object g at time k differs from the estimated track ID most recently associated with that object.

To compute IDF1, an overlap matrix is first formed for each sequence from the number of object frames in which a ground-truth track ID and an estimated track ID are simultaneously matched. The Hungarian algorithm establishes a one-to-one assignment between the two ID sets that maximizes the total overlap. The overlap sum of the assigned ID pairs is defined as IDTP. Estimated object frames not included in IDTP are IDFP, while ground-truth object frames not included in IDTP are IDFN. IDF1 is then

$$\text{IDF1} = \frac{2 \text{IDTP}}{2 \text{IDTP} + \text{IDFP} + \text{IDFN}}. \quad (39)$$

The IDTP, IDFP, and IDFN values are summed over all evaluation sequences before the final IDF1 is computed. Lower IDSW and higher IDF1 indicate better identity-preservation performance.

B. Quantitative Performance Comparison

Table II compares one-step prior prediction accuracy. PR-IMM achieves the lowest error among all methods. This result indicates that the general physics-based motion models compensate when the data-driven PR prediction becomes unstable, while the mode probabilities automatically assign greater weight to the mode that best explains the current motion.

TABLE II
ONE-STEP PRIOR PREDICTION RMSE.

Model	RMSE _x ↓ [m]	RMSE _y ↓ [m]	RMSE _{xy} ↓ [m]
CV	4.884	3.801	6.189
CA	2.241	2.068	3.050
CT	4.872	3.545	6.025
PR	0.971	0.822	1.272
IMM (CV+CA+CT)	1.827	1.685	2.485
PR-IMM (Ours)	0.799	0.698	1.062

RadarScenes does not provide labels for motion scenarios, so scenario categories are derived from the previous ten ground-truth positions, timestamps, and the object's heading. Objects that satisfy none of the scenario conditions are excluded from scenario-wise evaluation. As shown in Table III, PR-IMM obtains the lowest RMSE_{xy} in all four motion scenarios. In particular, it outperforms the standalone PR model even in the Nonlinear scenario, which is difficult to explain with a single motion assumption. This demonstrates that combining data-driven prediction with physics-based motion models remains effective for complex motion.

TABLE III
SCENARIO-WISE PRIOR PREDICTION RMSE.

Scenario	CV	CA	CT	PR	IMM	Ours
Straight	4.007	1.308	5.119	0.633	1.038	0.584
Accel./Decel.	8.387	2.323	8.694	1.029	2.099	0.984
Left/Right Turn	7.055	2.999	6.781	1.024	2.291	0.904
Nonlinear	7.433	3.896	7.164	1.270	3.143	1.106

Table IV reports overall IDSW and IDF1. Compared with the standalone PR model, PR-IMM reduces IDSW from 915 to 519 and increases IDF1 from 0.8795 to 0.9020. This shows that the proposed method improves not only position RMSE but also the ability to preserve track IDs when multiple objects are close to one another.

TABLE IV
OVERALL MULTIPLE-OBJECT-TRACKING PERFORMANCE (GT OBJECT FRAMES = 17,866).

Model	IDSW ↓	IDF1 ↑
CV	954	0.7583
CA	700	0.8075
CT	901	0.7571
PR	915	0.8795
IMM (CV+CA+CT)	695	0.8228
PR-IMM (Ours)	519	0.9020

To analyze tracking performance under more difficult conditions, we additionally construct scenarios in which data association becomes ambiguous because objects are spatially close or their paths cross. Table V summarizes these results. Close Proximity denotes frames in which the nearest object is within 4 m. Crossing Motion denotes cases in which two objects are within 8 m and their heading difference is between 45° and 135°. Missed-detection scenarios are generated by removing 10% or 20% of the measurements in each sequence using the same random seed. PR-IMM achieves the lowest IDSW and the highest IDF1 in every scenario. Thus, it remains robust even when data association is more ambiguous or observations are partially missing.

TABLE V
SCENARIO-WISE MULTIPLE-OBJECT-TRACKING PERFORMANCE.

Scenario (GT object frames)	Metric	PR	IMM	PR-IMM (Ours)
Close Proximity (1,284)	IDSW↓	54	43	34
	IDF1↑	0.8769	0.8100	0.8941
Crossing Motion (687)	IDSW↓	20	21	11
	IDF1↑	0.9127	0.8486	0.9258
10% Miss (17,866)	IDSW↓	1,245	836	711
	IDF1↑	0.7316	0.7185	0.7610
20% Miss (17,866)	IDSW↓	1,952	1,197	1,146
	IDF1↑	0.5476	0.5629	0.5849

C. Qualitative Analysis

This subsection qualitatively analyzes PR-IMM through mode-probability statistics and trajectory visualizations.

1) *Mode Probability Visualization*: Figure 4 shows the average probability of each PR-IMM mode over the entire test dataset. PR has the highest mean probability at 0.348, followed by CA at 0.304, CV at 0.185, and CT at 0.162. PR receives the largest share because the Transformer predictor frequently provides a prior state close to the observation based on the recent trajectory and therefore tends to obtain a relatively high likelihood.

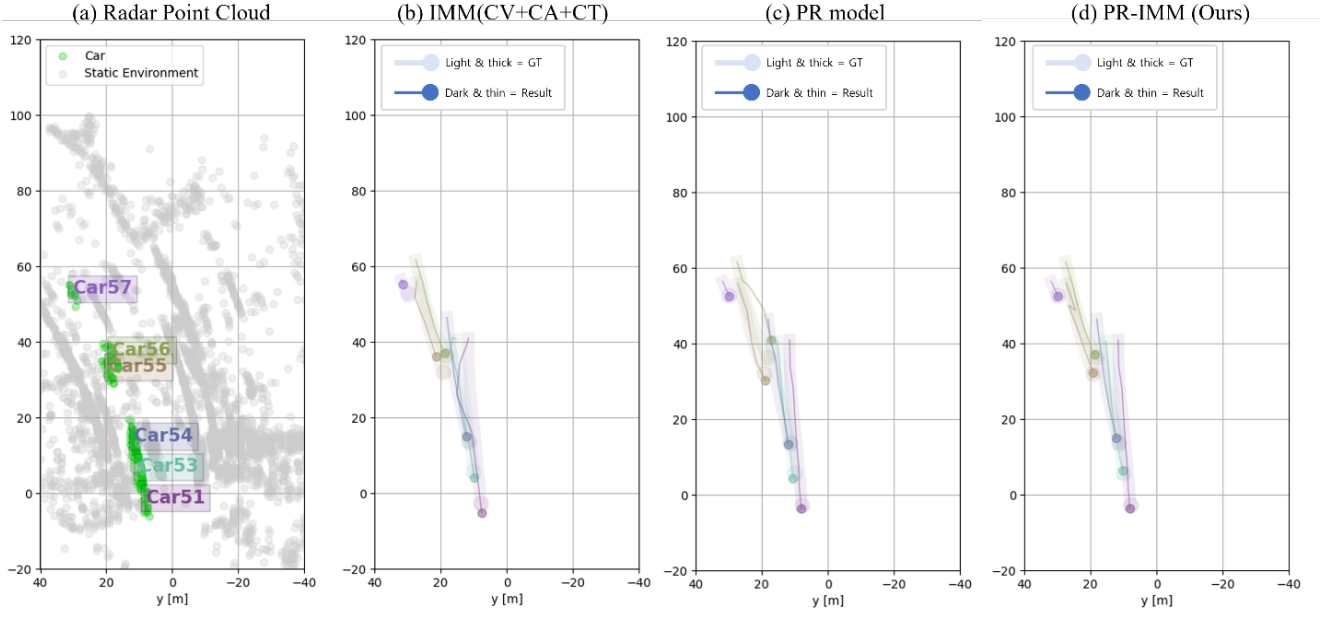


Fig. 3. Qualitative tracking results on the RadarScenes dataset. (a) Radar point cloud with annotated vehicle points. (b) CV/CA/CT-based IMM result. (c) PR model result. (d) Proposed PR-IMM result. In (b)–(d), light thick lines denote ground-truth trajectories, whereas dark thin lines denote estimated trajectories.

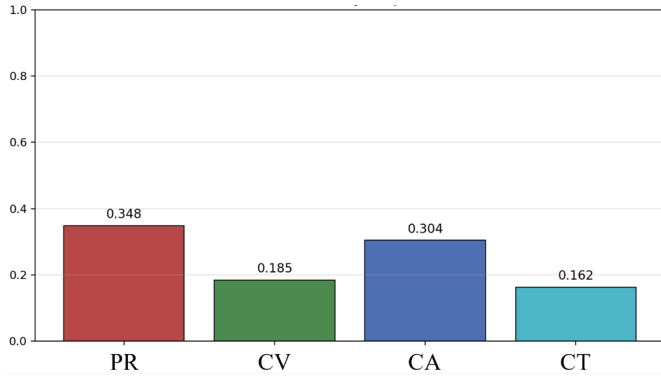


Fig. 4. Mean mode probability of PR-IMM.

However, the PR probability is only 0.348 and is therefore not dominant by itself; more than half of the total probability is distributed among CV, CA, and CT. The CA mode has a higher average probability than CV and CT, suggesting that the dataset contains a substantial number of accelerating and decelerating intervals in which CA complements the PR prediction. These results support the reason why PR-IMM outperforms PR alone: tracking does not rely on a single mode, but rather combines the data-driven predictor and physics-based motion models in a complementary manner.

2) *Object Trajectory Visualization*: Figure 3 presents qualitative object-tracking results. The conventional IMM (CV+CA+CT) shows the largest deviations of multi-object trajectories from the ground truth, with particularly noticeable dispersion where objects are close to one another. The PR model follows the ground truth more closely than the conventional IMM, but its estimated positions still fluctuate in some intervals. In contrast, PR-IMM produces trajectories that most closely match the ground truth throughout the sequence and tracks each object stably even when multiple objects are adjacent. These observations are consistent with the RMSE results and demonstrate that PR-IMM robustly combines data-

driven and physics-based prior states in both quantitative and qualitative evaluations.

V. CONCLUSION

This paper proposed a PR-IMM tracker for radar-point-cloud-based multiple-object tracking. The PR model converts radar observations into relative displacements with respect to the most recent observation and uses a Transformer encoder to predict the next displacement, allowing nonlinear object motion to be learned directly from data without a fixed motion assumption. The PR model is then integrated as an independent IMM mode together with CV, CA, and CT. Because the four modes have different state dimensions, input mixing and state fusion are performed using the two-dimensional position and position covariance shared by all modes, enabling data-driven and physics-based prior states to be dynamically combined according to the current motion situation.

Experiments show that PR-IMM achieves the best performance among the CV, CA, and CT single-motion models, the conventional IMM (CV+CA+CT), and the standalone PR model. Analysis of mode probabilities and tracking trajectories further confirms the complementary operation of the PR mode and the physics-based modes. These results demonstrate that integrating a learned state predictor into the probabilistic IMM framework can improve both position-estimation accuracy and identity-preserving multiple-object tracking under diverse motion and observation conditions.

ACKNOWLEDGMENT

This work was supported by the Ministry of SMEs and Startups, Republic of Korea, in 2026 under Grant RS-2025-24535910.

REFERENCES

- [1] G. Han and S.-C. Kee, "A hybrid approach multi-object tracking with optical flow and kalman filter," *Transactions of the Korean Society of Automotive Engineers*, vol. 33, no. 9, pp. 755–762, 2025.

- [2] D. Yoon, K. Jang, J. Won, and Y. Kang, "A multi-task deep learning method for road environment recognition and object tracking," *Transactions of the Korean Society of Automotive Engineers*, vol. 32, no. 1, pp. 77–82, 2024.
- [3] H.-Y. Jung, D.-H. Paek, and S.-H. Kong, "Open-source autonomous driving software platforms: Comparison of Autoware and Apollo," *arXiv preprint arXiv:2501.18942*, 2025.
- [4] S. J. Cho, B. S. Kim, T. S. Kim, and S.-H. Kong, "Enhancing GNSS performance and detection of road crossing in urban area using deep learning," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 2115–2120.
- [5] S.-H. Kong and B. Kim, "Two-dimensional compressed correlator for fast PN code acquisition," *IEEE Transactions on Wireless Communications*, vol. 12, no. 11, pp. 5859–5867, 2013.
- [6] S.-H. Kong, "Fast multi-satellite ML acquisition for A-GPS," *IEEE Transactions on Wireless Communications*, vol. 13, no. 9, pp. 4935–4946, 2014.
- [7] S.-S. Jan and Y.-C. Kao, "Radar tracking with an interacting multiple model and probabilistic data association filter for civil aviation applications," *Sensors*, vol. 13, no. 5, pp. 6636–6650, 2013.
- [8] A. Manjunath, Y. Liu, B. Henriques, and A. Engstle, "Radar based object detection and tracking for autonomous driving," in *2018 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*, 2018, pp. 1–4.
- [9] H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with markovian switching coefficients," *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, 1988.
- [10] W. Lv, Y. Huang, N. Zhang, R. S. Lin, M. Han, and D. Zeng, "Diffmot: A real-time diffusion-based multiple object tracker with non-linear prediction," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 19 321–19 330.
- [11] D.-I. Kim, D.-H. Paek, S.-H. Song, and S.-H. Kong, "Bayesian approximation-based trajectory prediction and tracking with 4d radar," in *2025 IEEE Intelligent Vehicles Symposium (IV)*, 2025, pp. 1380–1385.
- [12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 5998–6008.
- [13] M. Kim, B. I. Kwak, J. U. Hou, and T. Kim, "Robust long-term vehicle trajectory prediction using link projection and a situation-aware transformer," *Sensors*, vol. 24, no. 8, p. 2398, 2024.
- [14] X. Tang, X. Cheng, and N. Xu, "A robust multiobject tracking method based on 4-d millimeter-wave radar and monocular vision fusion," *IEEE Sensors Journal*, vol. 24, no. 22, pp. 37 764–37 774, 2024.
- [15] O. Schumann, M. Hahn, N. Scheiner, F. Weishaupt, J. F. Tilly, J. Dickmann, and C. Wöhler, "Radarscenes: A real-world radar point cloud data set for automotive applications," in *2021 IEEE 24th International Conference on Information Fusion (FUSION)*, 2021, pp. 1–8.
- [16] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3464–3468.
- [17] Y. Li, C. Liang, M. Lu, X. Hu, and Y. Wang, "Cascaded kalman filter for target tracking in automotive radar," *The Journal of Engineering*, vol. 2019, no. 19, pp. 6264–6267, 2019.
- [18] Q. Li, Z. Zhang, G. He, X. Hu, and X. Kang, "3d multi-object tracking for autonomous driving based on IMM-EKF and re-identification," in *2024 IEEE International Conference on Unmanned Systems (ICUS)*, 2024, pp. 1197–1202.
- [19] X. Liu, X. Zhao, G. Liu, Z. Wu, T. Wang, L. Meng, and Y. Wang, "IMM-MOT: A novel 3d multi-object tracking framework with interacting multiple model filter," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025, pp. 20 540–20 547.
- [20] T. Meinhardt, A. Kirillov, L. Leal-Taixe, and C. Feichtenhofer, "Trackerformer: Multi-object tracking with transformers," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 8844–8854.
- [21] P. Wang, Y. Wang, H. Cao, W. Chen, and D. Li, "LA-MOTR: End-to-end multi-object tracking by learnable association," in *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025, pp. 12 438–12 448.
- [22] L. Cheng, A. Sengupta, and S. Cao, "Deep learning-based robust multi-object tracking via fusion of mmwave radar and camera sensors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 11, pp. 17 218–17 233, 2024.
- [23] Y. Bai, B. Yan, W. Dong, X. Jin, T. Su, and H. Ma, "A novel adaptive state estimation model: Kalman filter coupled with neural networks," *International Journal of Adaptive Control and Signal Processing*, vol. 39, no. 5, pp. 914–926, 2025.
- [24] S. Kim, J. Choi, D. Jeong, and S. Lee, "LSTM-based kalman filter error compensation for enhanced moving object tracking," *Journal of Institute of Control, Robotics and Systems*, vol. 31, no. 1, pp. 68–76, 2025.
- [25] S.-H. Kong, D.-H. Paek, and S. Lee, "RTNH+: Enhanced 4d radar object detection network using two-level preprocessing and vertical encoding," *IEEE Transactions on Intelligent Vehicles*, vol. 10, no. 2, pp. 1427–1440, 2025.
- [26] Z. Pan, F. Ding, H. Zhong, and C. X. Lu, "Ratrack: Moving object detection and tracking with 4d radar point cloud," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 4480–4487.