Hybrid Machine Learning and Estimation-based Flight Trajectory Prediction in Terminal Airspace

HONG-CHEOL CHOI, CHUHAO DENG, AND INSEOK HWANG, (Member, IEEE)
School of Aeronautics and Astronautics, Purdue University, West Lafayette, Indiana 47907 USA
Corresponding author: Hong-Cheol Choi (e-mail: choi642@purdue.edu).
This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant 21BDAS-B158275-02).

ABSTRACT
For air traffic management, trajectory prediction plays an important role as the predicted trajectory information is used in crucial tasks for the safety and efficiency of air traffic operations, such as conflict detection and resolution, scheduling, and sequencing. In this paper, we propose a framework for trajectory prediction in terminal airspace by combining a machine learning-based method and a physics-based estimation method. A trajectory prediction model based on machine learning is trained from historical surveillance data to represent the collective behavior of a set of flight trajectories, from which the data-driven prediction can be obtained as the expected future behavior of an incoming flight. A physics-based estimation algorithm called Residual-Mean Interacting Multiple Models (RM-IMM) then incorporates the machine learning prediction as a pseudo-measurement to account for the current motion of the aircraft. The proposed framework is tested, with real air traffic surveillance data, by predicting the future state information of the flights for real-time air traffic control applications. The results show that the proposed framework produces a greatly improved prediction accuracy compared to the two existing machine learning-based algorithms.

INDEX TERMS
Aircraft trajectory prediction, terminal airspace, machine learning, Gaussian mixture model, long short-term memory network, residual-mean interacting multiple models.

I. INTRODUCTION
To accommodate the growing demand of air traffic, the modern Air Traffic Management (ATM) system becomes one of the most complex and vast systems. Among the operations in airspace which consists of en-route, terminal, and surface, terminal airspace operations have a higher impact on the entire ATM system’s safety and efficiency due to the high traffic density and highly structured operational procedures, such as Standard Terminal Arrival Route (STAR) and Standard Instrument Departure (SID). According to the data published by Boeing [1], 65 percent of fatal accidents occur in terminal airspace, while it accounts for only 6 percent of the total flight time. Therefore, most research on decision support tools has focused on terminal airspace to help Air Traffic Controllers (ATC) who are responsible for the safety and efficiency in terminal airspace. The performance of decision support tools highly relies on the accuracy of trajectory prediction for ATC’s tasks such as conflict detection and resolution, sequencing, and scheduling. In this regard, there have been extensive efforts in the development of trajectory prediction algorithms, which can be categorized as estimation-based methods and machine learning-based methods.

In estimation-based methods, the behaviors of aircraft are characterized based on the governing dynamics, and the aircraft’s current states are propagated into the future timesteps based on the dynamics by using tools such as Kalman filtering or its variations. In [2], a trajectory prediction technique is proposed based on a stochastic differential equation that describes the aircraft’s motion disturbed by wind, and the distribution of solutions to the stochastic differential equation is approximated as a Markov chain. In [3], the aircraft’s dynamics is represented by using a stochastic linear hybrid system (SLHS). A hybrid estimation technique is combined with an intent inference algorithm for aircraft trajectory prediction. This approach has been extended in [4] by considering a state-dependent transition model to better represent the changes in flight modes, such as Constant Velocity (CV) and Coordinated Turn (CT) modes.
In [5], an online four-dimensional trajectory prediction technique is developed based on a point-mass model and aircraft performance data, along with a conformance monitoring and aircraft intent update process for more accurate trajectory prediction.

In machine learning-based methods, a trajectory pattern is identified as a set of flight trajectories with similar behaviors, from which a data-driven model is trained to represent the collective behavior of the trajectory pattern using machine learning techniques such as support vector regression [6], Gaussian Mixture Models (GMM) [7], [8], recurrent neural network [9], Long Short-Term Memory (LSTM) [10], [11], and hybrid deep learning [12]–[14]. With the trained model, the future trajectory is predicted as the expected behavior of the aircraft in the future timesteps, given a series of measurements observed until the current timestep. For specific operational conditions, several machine learning-based methods have been developed for trajectory prediction in en-route airspace [11], [13], [15]–[18] and terminal airspace [7], [8], [10], [12], [19]. Especially in terminal airspace, [8], [19] focused on the operations in extended terminal airspace (about 20 minutes or 100 nm from an airport) to predict the Estimated Time of Arrival (ETA) of arrival flights, while [7], [10] presented the prediction of aircraft’s position in the airspace close to an airport (1 to 2.5 minutes from an airport), for both takeoff and landing. In [19], major trajectory patterns are first identified by using a clustering method and then the ETA for each pattern is learned based on a regression model. The ETA of an aircraft is obtained as a weighted sum of ETAs of individual trajectory patterns, where the weight is the probability of the aircraft following that specific trajectory pattern. In [7], a trajectory prediction is performed for 1 minute from an airport using a probabilistic trajectory model based on the GMM, which is also used in [8] for the extended terminal airspace, considering the different time horizons up to 100 nm from an airport. A deep learning-based approach is proposed in [10] based on LSTM for the prediction horizon of 1.5 minutes for takeoff and 2.5 minutes for landing. In [20], multi-aircraft trajectory prediction is proposed based on the social LSTM by effectively capturing the interaction between aircraft.

In the estimation-based methods, the expected future behaviors of an aircraft rely on the assumption that the aircraft follows its flight plan or an a priori known trajectory pattern. In this paper, such information is extracted from historical data by using machine learning techniques that can represent the collective behavior (or pattern) of the aircraft which operated in terminal airspace. Note that there could be some degree of errors in the trained model due to the assumed structure of the machine learning model, and the machine learning-based methods cannot explicitly account for the current motion of an aircraft. Hence, the machine learning model could generate inaccurate (possibly unfeasible) future states that violate the aircraft dynamics. To make up for it, we propose a flight trajectory prediction framework by combining the following two approaches: The collective behavior of a set of similar flight trajectories is represented as a machine learning model trained from historical data; and for the aircraft’s observed states until the current timestep, the machine learning model is then used to generate the expected states in the future timesteps, which are fed into an estimation-based method that combines the expected states from the data with the propagated states from the current state into the future timesteps based on the aircraft dynamics. The framework of the proposed method is illustrated in Fig. 1 which consists of (i) data preparation by using a pattern identification framework, (ii) construction of a machine learning model with the identified trajectory patterns, and (iii) the hybrid trajectory prediction method that combines the machine learning-based method and the estimation-based method. The main contributions of this paper are as follows:

- A novel trajectory prediction method based on combined GMM and Residual-Mean Interacting Multiple Models (RM-IMM) [21] is developed. GMM is used for mining the future pseudo measurements based on the past measurements until the current timestep. RM-IMM
is used for inferring the current dynamics of aircraft and estimating the future trajectory. To the best of our knowledge, it is the first work to combine a machine learning-based method and an estimation-based method to enhance aircraft trajectory prediction performance in complex terminal airspace.

- The proposed method guarantees the real-time prediction of trajectory for 2 minutes look-ahead time, while the combined LSTM and RM-IMM method failed to predict in real-time. This prediction horizon is set by considering the response time of both Conflict Alerts (CAs) and Minimum Safe Altitude Warnings (MSAWs) in literature [22].
- The proposed trajectory prediction method generates a greatly improved prediction accuracy compared to the baseline algorithms such as LSTM.

For the rest of this paper, the data preparation procedure is presented in Section II, and the proposed framework for trajectory prediction is described in Section III, followed by the demonstration with the real data in Section IV. Concluding remarks are given in Section V.

II. DATA PREPARATION

The air traffic surveillance data used in this paper are collected from the Automatic Dependent Surveillance-Broadcast (ADS-B) system. The dataset provides the aircraft’s states, such as timestamp, position (longitude, latitude, altitude), and speed (horizontal and vertical) for each flight trajectory. In this paper, we consider the arrival and departure flights around the major airport in South Korea, Incheon International Airport (ICN), from January to August in 2019. The flight trajectories during a single month (January 2019) are shown in Fig. 2. During the period of January to August 2019, the number of arrival and departure trajectories are identified as 130,110 and 138,999, respectively. The track points of each trajectory are recorded every minute but with some missing or repeated timestamps. The trajectories with heavily missing points are excluded from the dataset and repeated points are removed. After performing data cleaning, the trajectories recorded within 60 nm from the airport are considered as flights in the entire terminal airspace of ICN. To handle the trajectories more effectively and efficiently, the trajectories with similar behaviors are grouped together by using the trajectory pattern identification algorithm that we developed [23]. Based on this framework, firstly, the dissimilarity between trajectories is computed using Dynamic Time Warping (DTW) [24] and Euclidean distance. Then, trajectories are being linked using the Ward’s linkage method [25] and a dendrogram can be built. The dendrogram allows us to choose the number of trajectory patterns that we want. The number of trajectory pattern is chosen such that all trajectories within one pattern share the same set of waypoints.

As an illustrative example, the resultant trajectory patterns, a set of the trajectories arriving at and departing from ICN along two specific routes, are presented in Fig. 3. The arrival flights pass through a fix at the east of ICN, called KARBU, which is the entry fix to the arrival route, and then approach from the southeast direction for landing (i.e., landing at Runway 33L/R or 34). Conversely, the departure flights take off from Runway 15L/R and pass through a fix at the east of ICN, called EGOBA, which is the last fix of the departure route. For the use of the data in the downstream step of the proposed trajectory prediction framework, the flight trajectories are resampled to have the same sampling time ($\Delta t$) and normalized between 0 and 1 in order to be fed into machine learning models. The normalization is computed as:

$$\hat{X} = \frac{X - \text{min}}{\text{max} - \text{min}}$$

where $X$ is a flight trajectory data, $\text{max}$ and $\text{min}$ indicate the maximum and the minimum value of the sample, respectively, and $\hat{X}$ is the normalized data.
III. METHODOLOGY

In this section, we present the proposed algorithm for trajectory prediction in terminal airspace, that is, the hybrid machine learning and estimation-based trajectory prediction method. Two machine learning-based prediction methods are firstly explained as baseline algorithms: (i) conventional Gaussian Mixture Model (GMM) and (ii) Long Short-Term Memory (LSTM) which is a widely used deep learning method for time-series data. A Stochastic Linear Hybrid System (SLHS) is introduced and then the proposed hybrid approach for trajectory prediction is presented by combining a machine learning model and an estimation method.

A. MACHINE LEARNING-BASED PREDICTION METHODS

1) GMM-based Trajectory Prediction Method

Each trajectory pattern has similar property along the spatial dimension, however, there exists some degree of variability in not only the spatial but also temporal dimensions. To capture such variability in each trajectory pattern, we use a GMM [26], that is, the probability distribution of datapoints can be modeled as a mixture of multiple Gaussian distributions. In the air traffic surveillance data, a flight trajectory is given in the form of a temporal sequence:

\[ X = [x_0, x_1, \cdots, x_T]^T \]  \hspace{1cm} (2)

where \( x_t \) is the aircraft’s position (longitude and latitude in the horizontal dimension and altitude in the vertical dimension) at timestep \( t \in [0, 1, \cdots, T] \) and \( T \) is the final timestep. The set of \( N \) flight trajectories is used to train a GMM whose Probability Density Function (PDF) is given as:

\[ p(X) = \sum_{k=1}^{K} w_k \mathcal{N}(X | \mu_k, \Sigma_k) \]  \hspace{1cm} (3)

where \( \mathcal{N}(X | \mu_k, \Sigma_k) \) is the PDF of the multivariate Gaussian distribution with the mean \( \mu_k \) and the covariance \( \Sigma_k \), and \( w_k \in [0, 1] \) is the mixing coefficient for the \( k \)-th Gaussian component \( \{k \in [1, \cdots, K]\} \) which sums up to 1, i.e., \( \sum_{k=1}^{K} w_k = 1 \). For learning the GMM, we use the Expectation-Maximization (EM) algorithm which finds a maximum-likelihood fit for the given data. The number of Gaussian components, \( K \), is a design parameter for balancing underfitting (with a smaller \( K \), which leads to a large training error) and overfitting (with a larger \( K \), which leads to a large test error), which can be determined by using the criterion such as Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC) [27], or by investigating the performance measure such as the prediction error.

For a new incoming flight observed up to the current timestep, the learned GMM corresponding to a trajectory pattern can generate its future trajectory for a given time horizon. We define a prediction time, \( T_p \), as the timestep when the prediction is performed (or the current timestep) and a prediction horizon, \( T_h \), as the duration of the future timesteps during which the future trajectory is predicted \( (T_p + T_h \leq T) \). Let each trajectory be partitioned as the past part \( X^P = [x_0, x_1, \cdots, x_{T_p}]^T \) and the future part \( X^F = [x_{T_p+1}, \cdots, x_{T_p+T_h}]^T \). Similarly, the mean and the covariance of a Gaussian component are partitioned as:

\[
\mu_k = \begin{bmatrix} \mu_k^P \\ \mu_k^F \end{bmatrix}, \quad \Sigma_k = \begin{bmatrix} \Sigma_k^{PP} & \Sigma_k^{PF} \\ \Sigma_k^{FP} & \Sigma_k^{FF} \end{bmatrix}
\]  \hspace{1cm} (4)

For a new incoming flight, \( X^P_{\text{new}} \), its posterior probability for the \( k \)-th Gaussian component, \( k \in [1, \cdots, K] \), is computed as:

\[
\tilde{w}_k = \frac{w_k \mathcal{N}(X^P_{\text{new}} | \mu_k, \Sigma_k)}{\sum_{j=1}^{K} w_j \mathcal{N}(X^P_{\text{new}} | \mu_j, \Sigma_j)}
\]  \hspace{1cm} (5)

The conditional mean and covariance of \( X^F_{\text{new}} \) with respect
to the $k$-th Gaussian component are then given as:
\[
\begin{align*}
\tilde{\mu}_k &= \mu_k^F + \Sigma_k^F (\Sigma_k^P)^{-1} (X_{new}^P - \mu_k^P) \\
\tilde{\Sigma}_k &= \Sigma_k^F - \Sigma_k^F (\Sigma_k^P)^{-1} \Sigma_k^F
\end{align*}
\]
Finally, the PDF of $X_{new}^F$, conditioned on $X_{new}^P$ is obtained as:
\[
\sum_{k=1}^{K} \tilde{w}_k \mathcal{N}(X_{new}^F | \tilde{\mu}_k, \tilde{\Sigma}_k)
\]
which yields the total conditional mean (or the predicted trajectory) as:
\[
\tilde{X}_{new}^F = \sum_{k=1}^{K} \tilde{w}_k \tilde{\mu}_k
\]

For the illustration, we demonstrate two prediction examples as shown in Fig. 4. Given the number of Gaussian components of $K = 10$ and 10 past measurements (black star), the GMM computes the means from the posterior distribution (line) and predicts the future states (red) of the aircraft using Eqs. (6) and (9).

2) LSTM-based Trajectory Prediction Method
In order to effectively process temporal sequential data in Eq. (2) using deep learning, the Recurrent Neural Network (RNN) [29], or its variant, called Long Short-Term Memory (LSTM) [30], has been widely used in flight trajectory prediction [10]–[13], [15], [31] to utilize its power in memorizing the information in the previous timesteps. To generate a one-step ahead prediction using LSTM by taking all the previous measurements, the input time-series and output time-series are constructed with a one-step shift, that is, for each trajectory, the input and output are given as $\{x_t\}_{t=0}^{T-1}$ and $\{y_t\}_{t=0}^{T-1}$ where $y_t = x_{t+1}$, respectively.

The structure of LSTM is shown in Fig. 5 where $x_t$ is an input element and $\hat{y}_t$ is a predicted output element at timestep $t$. In the computation of $\hat{y}_t$, the temporal dependency of the input elements in the previous timesteps, from 0 to $t-1$, can be incorporated by the hidden state $h_{t-1}$ and the memory cell $c_{t-1}$. At timestep $t$, the memory cell $c_t$ is first updated as:
\[
c_t = \Gamma_f c_{t-1} + \Gamma_u \tilde{c}_t
\]
where $\tilde{c}_t$ is the pseudo-memory cell at timestep $t$ to update the new information from $x_t$, which is fused with the memory cell at the previous timestep, $c_{t-1}$, to obtain the memory cell at timestep $t$, $c_t$. The information flow is controlled by the forget gate, $\Gamma_f \in [0, 1]$ (for the past information, $c_{t-1}$), and the update gate, $\Gamma_u \in [0, 1]$ (for the new information, $\tilde{c}_t$). These gates including the output gate, $\Gamma_o \in [0, 1]$, are given as:
\[
\Gamma(\cdot) = \sigma (W(\cdot)_h h_{t-1} + W(\cdot)_x x_t + b(\cdot))
\]
for $(\cdot) \in \{f, u, o\}$
where $\sigma (\cdot)$ is the sigmoid activation function and the parameters $W$'s and $b$'s are the weight and bias parameters, respectively. The hidden state, $h_t$, and the predicted output,
where \(\zeta(t+1) = z(t+1)\) and \(R(t+1) = R(t+1)\), the mode-conditioned continuous state estimates and mode probabilities are updated as follows:

- In each discrete state (or mode) \(j \in \mathbb{Q}\), the mode-conditioned continuous state estimate \(\hat{\xi}_j(t+1)\) and its covariance \(P_j(t+1)\) are updated by a mode-matched Kalman filter:
  \[
  \hat{\xi}_j(t+1) = A_j \hat{\xi}_{0j}(t) + K_j(t+1) \nu_j(t+1)
  \]
  \[
  P_j(t+1) = [I - K_j(t+1)C_j] \hat{P}_j(t+1)\|
  \]

where \(\hat{\xi}_{0j}(t)\) is the initial condition of the state estimate, \(K_j(t+1)\) is the Kalman filter gain, and \(\nu_j(t+1)\) is the innovation computed by Kalman filter \(j\). \(P_j(t+1)\) is the covariance of the state estimate at timestep \(t+1\) before the measurement update.

- The mode probability \(\alpha_j(t+1)\) is updated based on the Bayesian rule for all \(j \in \mathbb{Q}\):
  \[
  \alpha_j(t+1) = \frac{1}{m(t+1)} A_j(t+1) \sum_{i=1}^{n_d} \pi_{ij} \alpha_i(t)
  \]

where \(m(t+1)\) is a normalization constant and \(A_j(t+1)\) is the likelihood function of mode \(j\) computed with the residual and its covariance by Kalman filter \(j\).

The output of the algorithm is obtained as the combined continuous state estimate in the form of a Gaussian PDF, \(\mathcal{N}(\hat{\xi}(t+1), P(t+1))\) with mean \(\hat{\xi}(t+1)\) and covariance

\[
\Pi = [\pi_{ij}]_{i,j \in \mathbb{Q}}
\]
In order to integrate the machine learning model and the estimation algorithm, we propose to use the machine learning prediction as a pseudo measurement for an estimation-based prediction using RM-IMM, to account for the expected future behavior as well as the current motion of an aircraft. As shown in Fig. 7, the trained model first generates a pseudo measurement,  \( \tilde{z}(t+1) \), from which the PDF of the one-step ahead prediction can be obtained as \( \mathcal{N}(\hat{\tilde{z}}(t+1), \hat{\tilde{R}}(t+1)) \) where

\[
\hat{\tilde{z}}(t + 1) = C \hat{\xi}(t + 1) \tag{23}
\]

\[
\hat{\tilde{R}}(t + 1) = CP(t + 1)C^T + \hat{\tilde{R}}(t + 1) \tag{24}
\]

using Eq. (16).

2) Integration of Machine Learning-based and Estimation-Based Methods

In order to integrate the machine learning model and the estimation algorithm, we propose to use the machine learning prediction as a pseudo measurement for an estimation-based prediction using RM-IMM, to account for the expected future behavior as well as the current motion of an aircraft. As shown in Fig. 7, the trained model first generates a pseudo measurement by taking the previous measurements up to timestep \( t \). The estimation algorithm (RM-IMM) propagates the current state through the aircraft dynamics and then corrects the propagated state by using the pseudo measurement.

Note that the input and output of the estimation-based method are given in a stochastic form, i.e., the PDF of the data-driven pseudo measurement, \( \mathcal{N}(\tilde{z}(t+1), \tilde{R}(t+1)) \), as input and the PDF of the (posterior) prediction, \( \mathcal{N}(\hat{z}(t+1), \hat{R}(t+1)) \), as output. Since this information can be readily provided by the learned GMM using Eq. (8), the incorporation of the GMM-based predicted measurement and RM-IMM can be achieved without additional integration processes. However, the LSTM-based method works in a deterministic manner, that is, it takes a sequence of points, \( x_0, \ldots, x_t \), and generates a pseudo measurement,  \( \tilde{z}(t+1) \). To address the mismatch in the format of the input and output, we integrate the LSTM model and RM-IMM as follows:

- From the output of RM-IMM given as a PDF at timestep \( t \), \( \mathcal{N}(\hat{z}(t), \hat{R}(t)) \), a set of points are sampled. Each sampled point is then fed into the LSTM-based prediction method to obtain an output point at timestep \( t+1 \).
- The set of output points from the LSTM-based prediction method are fitted into a Gaussian PDF, \( \mathcal{N}(\tilde{z}(t+1), \tilde{R}(t+1)) \), to be used as an input to RM-IMM.

Therefore, it is concluded that the proposed hybrid trajectory prediction approach can be applicable to conventional machine learning (GMM) as well as deep learning (LSTM), that is, hybrid GMM and RM-IMM and hybrid LSTM and RM-IMM are made available.

However, since the trajectory prediction should be achieved in an online fashion from the application perspective, the computational efficiency of the algorithm is crucial. We run four prediction algorithms on a computer with Intel Core i7-9750H CPU, 16 GB RAM, and NVIDIA GeForce RTX 2060 GPU under a Windows system. The computation time of predicting the flight trajectory for 2 minutes lookahead time is measured as shown in Table 1. The computation time of the LSTM-based hybrid method is significantly larger than that of the other algorithms due to the numerically intensive sampling process, which means it is unable to meet the requirement for real-time applications. For this reason, the GMM, LSTM model, and GMM-based hybrid method are used to perform experimental tests and validation in the next section.

### IV. EXPERIMENTAL RESULTS

To test and demonstrate the methodology described in Section III with the processed data in Section II, we compare and analyze three trajectory prediction methods by using historical ADS-B data in this section. The dataset has a total of 269,109 trajectories, which include 130,110 arrivals and 138,999 departures. The dataset for both GMM and LSTM is divided into a training dataset and a test dataset with the ratio 8:2, and 20% of the training dataset for LSTM is used as a validation dataset.

| Methods          | Mean (sec) | Standard deviation (sec) |
|------------------|------------|--------------------------|
| GMM              | 0.0085     | 0.0015                   |
| Hybrid (GMM)     | 0.1010     | 0.0054                   |
| LSTM             | 0.8890     | 0.0280                   |
| Hybrid (LSTM)    | 177.80     | 5.5988                   |

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3126117, IEEE Access
FIGURE 8. Illustrative prediction results of the three methods for arrival and departure flights.

(a1) Illustration of prediction for arrival flight.

(b1) Illustration of prediction for arrival flight.

(c1) Illustration of prediction for arrival flight.

(d1) Illustration of prediction for arrival flight.

(a2) Illustration of prediction for departure flight.

(b2) Illustration of prediction for departure flight.

(c2) Illustration of prediction for departure flight.

(d2) Illustration of prediction for departure flight.
A. PERFORMANCE METRICS

To measure the performance of the trajectory prediction methods in detail, we introduce four metrics that are widely used in trajectory prediction [33]–[35]. The horizontal error (HE) measures the difference between the actual and expected locations of an aircraft in the horizontal dimension. The along-track error (ATE) and cross-track error (CTE) measure the parallel and perpendicular difference to the actual course between the aircraft’s actual and predicted positions. The vertical error (VE) represents the altitude error in the vertical dimension. These metrics can also be computed by using Root Mean Square Error (RMSE) which is given by

\[
RMSE = \left( \frac{1}{n} \sum_{t=1}^{n} (P_t - R_t)^2 \right)^{\frac{1}{2}}
\]  

(25)

where \( n \) is the number of data points. \( P_t \) represents the trajectory predicted by a model and \( R_t \) denotes the actual trajectory at timestep \( t \). Since RMSE is always non-negative, the smaller the values of each metric, the closer the prediction is to the actual value, which means that the model’s prediction is more accurate.

B. COMPARATIVE ANALYSIS

The literature [22] has shown that the median response time (i.e., the time between the activation of an alert and the issue of a control instruction) following a Conflict Alerts (CAs) and a Minimum Safe Altitude Warnings (MSAWs) are 88 seconds and 38 seconds, respectively. In this regard, the 2-minutes-ahead prediction could help ATC’s situational awareness that is important for safe and efficient air traffic operations. In the experimental tests, the predictions of each method are performed over the horizon of 24 timesteps (120 seconds with the time interval \( \Delta t = 5 \) seconds).

For the illustration, we first present representative trajectories for the arrivals and the departures around ICN. Figure 8 presents a total of 4 prediction results of two arrivals (a, b) and two departures (c, d). The actual and the predicted trajectories in the horizontal dimension and in the vertical dimension are plotted in the left and the right, respectively. As can be seen from Fig. 8 (a1 - d1), the horizontal prediction of the three methods shows the same trend with the actual trajectory, but the predicted trajectory of the LSTM model deviates significantly, especially in the heading, from the actual trajectory compared with the other two methods. In Fig. 8 (a2 - d2), compared with the actual altitude, the predicted trajectory points of the LSTM model have large fluctuations at some timesteps. For all the plots in Fig. 8, it...
can generally be seen that the trajectories predicted by the proposed hybrid method are closest to the actual trajectory, with the smallest error, followed by GMM and LSTM.

To evaluate the performance of the proposed hybrid trajectory prediction method, the four metrics of HE, ATE, CTE, and VE discussed in Section IV-A are first computed based on the predicted trajectories of arrival aircraft and ground truth. For the illustration, the histograms of the metrics are presented in Fig. 9. The HE histogram of the proposed method is skewed to the left and the ATE, CTE, and VE histograms are concentrated around zero, which means the proposed method outperforms the other methods in terms of the given evaluation metrics.

We carried out the extensive trajectory prediction tests with all the available test dataset for the quantitative evaluation of the proposed method, and the RMSE of HE, ATE, CTE, and VE is used to measure prediction accuracy. The RMSE is computed using Eq. (25) and the results of arrival and departure flights are presented in Table 2 and Table 3, respectively. The comparison shows that the prediction errors, HE, ATE, CTE, and VE, of the proposed hybrid method for arrival flights are reduced by 46.8%, 48.6%, 24.2%, and 1.2% compared to the prediction errors of GMM on average and by 76.0%, 77.4%, 65.0%, and 55.8% compared to the prediction errors of the LSTM model on average. Similarly, the prediction errors of the hybrid method for departure flights also show significant performance improvements, that is, huge error reduction compared to the other two baseline algorithms, like the arrivals. Therefore, based on these quantitative analysis results, it is concluded that our hybrid trajectory prediction method outperforms the GMM and the LSTM model for the given time horizon.

In other words, the experimental results show that the GMM, as well as the proposed hybrid method, can predict the future position of the aircraft trajectories more accurately than the LSTM model in general. Interestingly, as shown in

### Table 2. RMSE results of a quantitative comparison between the three methods for arrival flights.

| Entry fix | Methods | HE (ft) | ATE (ft) | CTE (ft) | VE (ft) |
|-----------|---------|---------|----------|----------|--------|
| REBIT     | LSTM    | 4234.4  | 3514.4   | 1890.7   | 222.32 |
|           | GMM     | 1848.8  | 1529.3   | 604.39   | 147.45 |
|           | Hybrid  | 1054.5  | 855.96   | 436.33   | 141.43 |
| GUKDO     | LSTM    | 3144.4  | 2133.4   | 1805.6   | 210.37 |
|           | GMM     | 2085.1  | 1660.2   | 755.36   | 115.16 |
|           | Hybrid  | 1049.8  | 819.29   | 485.87   | 114.85 |
| OLMEN     | LSTM    | 4195.5  | 2133.4   | 1805.6   | 210.37 |
|           | GMM     | 2085.1  | 1660.2   | 755.36   | 115.16 |
|           | Hybrid  | 1049.8  | 819.29   | 485.87   | 114.85 |
| KARBUS    | LSTM    | 3790.3  | 1765.4   | 733.33   | 147.78 |
|           | GMM     | 2206.4  | 1765.4   | 733.33   | 147.78 |
|           | Hybrid  | 1238.3  | 995.33   | 485.87   | 114.85 |
| Average   | LSTM    | 4020.7  | 3579.8   | 994.39   | 290.16 |
|           | GMM     | 1814.5  | 1569.2   | 459.15   | 129.66 |
|           | Hybrid  | 964.61  | 806.41   | 348.12   | 128.12 |

### Table 3. RMSE results of a quantitative comparison between the three methods for departure flights.

| Last fix  | Methods | HE (ft) | ATE (ft) | CTE (ft) | VE (ft) |
|-----------|---------|---------|----------|----------|--------|
| EGOBA     | LSTM    | 3804.7  | 2410.7   | 2130.6   | 258.76 |
|           | GMM     | 2314.4  | 1710.0   | 898.88   | 141.65 |
|           | Hybrid  | 1375.2  | 1170.2   | 500.45   | 136.67 |
| OSPOT     | LSTM    | 3889.5  | 2410.7   | 2130.6   | 258.76 |
|           | GMM     | 2581.5  | 1994.8   | 945.10   | 181.77 |
|           | Hybrid  | 1498.3  | 1224.2   | 626.17   | 174.03 |
| BOPTA     | LSTM    | 4076.6  | 3525.8   | 1212.7   | 433.34 |
|           | GMM     | 1884.1  | 1588.5   | 364.73   | 154.53 |
|           | Hybrid  | 1161.7  | 1100.2   | 274.95   | 147.06 |
| BINIL     | LSTM    | 4752.5  | 3973.5   | 1176.3   | 358.06 |
|           | GMM     | 1868.9  | 1627.2   | 548.69   | 159.61 |
|           | Hybrid  | 954.96  | 911.24   | 359.95   | 152.59 |
| NOKPIK    | LSTM    | 4583.0  | 4269.1   | 1059.7   | 435.00 |
|           | GMM     | 2232.8  | 1791.6   | 655.36   | 168.14 |
|           | Hybrid  | 1330.0  | 1162.2   | 450.55   | 160.54 |
| Average   | LSTM    | 4136.7  | 3438.3   | 1028.3   | 330.42 |
|           | GMM     | 1814.5  | 1569.2   | 459.15   | 129.66 |
|           | Hybrid  | 964.61  | 806.41   | 348.12   | 128.12 |

FIGURE 10. An example of prediction error of the three methods over time.

FIGURE 11. Interaction between machine learning and estimation-based method.
Fig. 10, while the LSTM model performs similarly to the hybrid trajectory prediction method and better than the GMM for one or two-step prediction (corresponding to 5 or 10 seconds), it is quickly outperformed by the other two methods as the prediction time grows. This is because for LSTM, as discussed in Section III-A2, only the future position in one-step ahead can be predicted at a time, to achieve real-time prediction. For a look-ahead time of 2 minutes, the LSTM model needs to be implemented multiple times, each time with a new one-step prediction appended to the original data. Therefore, the error from each step’s prediction propagates, which causes the performance of the LSTM model to be worse than the proposed hybrid method and the GMM in multi-step trajectory prediction.

The prediction errors of the hybrid method are lower than those of GMM in general. The difference is attributed to the current dynamics (or flight mode) of the aircraft that is explicitly incorporated into the hybrid method, while the GMM generates future predictions based only on the learned model and past measurements. For the illustration, the prediction example by a single GMM that shows poor performance is presented in Fig. 11. The predicted trajectory of the GMM begins to slow down and turn as soon as the prediction starts, even though the aircraft maintains the Constant Velocity (CV) mode. This sudden change in the aircraft’s motion cannot be explained by its dynamics because the current motion of the aircraft follows the CV mode with a high probability. Due to the interaction between the GMM and RM-IMM in Fig. 7, the pseudo measurement from the GMM has been corrected by RM-IMM step by step, and thus the prediction is kept closer to the ground truth, while the prediction by a single GMM significantly deviates from the ground truth after five steps. Therefore, we conclude that the proposed hybrid machine learning and an estimation-based method can contribute to enhancing the prediction accuracy by facilitating the benefits of both methods.

V. CONCLUSION

In this paper, we proposed a framework for trajectory prediction in terminal airspace by combining a machine learning-based method and estimation-based method to enhance the prediction accuracy. The collective behavior of a trajectory pattern was represented as a machine learning model that is based on Gaussian Mixture Model (GMM), whose output serves as a pseudo measurement for an estimation-based prediction method, Residual-Mean Interacting Multiple Models (RM-IMM). The proposed method was tested and demonstrated with real air traffic surveillance data for the operations in the terminal airspace around Incheon International Airport (ICN) in South Korea. A total of 269,109 trajectories are considered in the experiments and the four metrics, horizontal error, along-track error, cross-track error, and vertical error, are used to measure the trajectory prediction errors. The quantitative comparison showed that the proposed method yields better accuracy than the GMM and LSTM model, which means the proposed method could help enhance ATC’s situational awareness that is important for the safety and efficiency of air traffic operations in terminal airspace.

However, the proposed method has the following limitations: (i) It considers 2 minutes look-ahead time based on the literature [22]. However, this work can be extended to longer prediction times; (ii) Reliability of data-driven trajectory prediction is not considered in this paper even though adversarial attack could induce the proposed method to generate incorrect predictions; and (iii) It used only surveillance data since other features such as meteorological data (i.e., wind speed and direction) and operational information are not included in the Automatic Dependent Surveillance-Broadcast (ADS-B) data. Therefore, in future work, these limitations will be further studied to achieve more accurate and reliable trajectory prediction for longer look-ahead times. In addition, we will enhance the prediction performance, which could be improved by performing a further clustering along the temporal dimension.

ACKNOWLEDGMENT

The authors are grateful to Dr. Hak-Tae Lee at Inha University for his valuable comments and support.

REFERENCES

[1] Airplanes, Boeing Commercial, “Statistical summary of commercial jet airplane accidents—worldwide operations 1959–2019,” Aviation Saf., 2020.
[2] J. Hu and M. Prandini. “Aircraft conflict detection: a method for computing the probability of conflict based on Markov chain approximation,” in 2003 European Control Conf. (ECC), IEEE, 2003.
[3] J.L. Yepes, I. Hwang, and M. Rotea. “New algorithms for aircraft intent inference and trajectory prediction,” J. Guid., Control, Dyn., 2007.
[4] W. Liu, C.E. Seah, and I. Hwang, “Estimation algorithm for stochastic linear hybrid systems with quadratic guard conditions,” in Proc. 48th IEEE Conf. Decision and Control held jointly with 2009 28th Chinese Control Conf., IEEE, 2009.
[5] J. Zhang, J. Liu, R. Hu, and H. Zhu, “Online four dimensional trajectory prediction method based on aircraft intent updating,” Aerosp. Sci. Technol., vol. 77, pp. 774–787, Jun. 2018.
[6] N. Xiao, P. Li, V. Hayvarichana, G.M. Hassana, D. Wang, and K. Li, “GOI: A novel design for vehicle positioning and trajectory prediction under urban environments,” IEEE Sensors Journal, pp. 5586–5594, 2018.
[7] S. T. Barratt, M. J. Kochenderfer, and S. P. Boyd, “Learning probabilistic trajectory models of aircraft in terminal airspace from position data,” IEEE Trans. Intell. Transp. Syst., vol. 20, no. 9, pp. 3536–3545, Sep. 2018.
[8] M.C.R. Murça and M.W. de Oliveira, “A Data-Driven Probabilistic Trajectory Model for Predicting and Simulating Terminal Airspace Operations,” in Proc. IEEE/AIAA 39th Digit. Avionics Syst. Conf. (DASC), 2020, pp. 1–8.
[9] P. Han, W. Wang, Q. Shi, and J. Yue, “A combined online-learning model with K-means clustering and GRU neural networks for trajectory prediction,” Ad Hoc Networks, 2021.
[10] W. Zeng, Z. Quan, Z. Zhao, C. Xie, and X. Lu, “A deep learning approach for aircraft trajectory prediction in terminal airspace,” IEEE Access, 2020.
[11] Y. Liu and M. Hansen, “Predicting aircraft trajectories: a deep generative convolutional recurrent neural networks approach,” arXiv preprint arXiv:1812.11670, 2018.
[12] L. Ma and S. Tian, “A hybrid CNN-LSTM model for aircraft 4D trajectory prediction,” IEEE Access, 2020.
[13] X. Zhang and S. Mahadevan, “Bayesian neural networks for flight trajectory prediction and safety assessment,” Decision Support Systems, 2020.
[14] N. Schimpf, E.J. Knoblock, Z. Wang, R.D. Apaza, and H. Li, “Flight Trajectory Prediction Based on Hybrid-Recurrent Networks,” in 2021 IEEE Cognitive Communications for Aerospace Applications Workshop, 2021.
[15] Z. Shi, M. Xu, Q. Pan, B. Yan, and H. Zhang, “LSTM-based flight trajectory prediction,” 2018 International Joint Conf. on Neural Networks (IJCNN), IEEE, 2018.

[16] Z. Chen, D. Guo, and Y. Lin, “A Deep Gaussian Process-Based Flight Trajectory Prediction Approach and Its Application on Conflict Detection,” Algorithms, no. 11, 2020.

[17] A. Bastas, T. Kravaris, and G.A. Vouros, “Data Driven Aircraft Trajectory Prediction with Deep Imitation Learning,” arXiv preprint arXiv:2005.07960, 2020.

[18] H. Paek, K. Lee, and A.E. Vela, “En-route Arrival Time Prediction using Gaussian Mixture Model,” in Proc. of the 9th International Conf. on Research and Air Transportation (ICRAT), Virtual Event, vol. 15, 2020.

[19] S. Hong and K. Lee, “Trajectory prediction for vectored area navigation arrivals,” Journal of Aerospace Information Systems, 12(7), 490-502, 2015.

[20] Z. Xu, W. Zeng, X. Chu, and P. Cao, “Multi-Aircraft Trajectory Collaborative Prediction Based on Social Long Short-Term Memory Network,” Aerospace, 2021.

[21] I. Hwang, H. Balakrishnan, and C. Tomlin, “State estimation for hybrid systems: applications to aircraft tracking,” IEE Proceedings-Control Theory and Applications, no. 5, 2006.

[22] K.R. Allendoerfer, S. Pai, and F.J. Friedman-Berg, “The complexity of signal detection in air traffic control alert situations,” in Proceedings of the Human Factors and Ergonomics Society Annual Meeting, vol. 52, no. 1, pp. 54-58. Sage CA: Los Angeles, CA: SAGE Publications, 2008.

[23] C. Deng, K. Kim, H. Choi, and I. Hwang, “Trajectory Pattern Identification for Arrivals in Vectored Airspace,” in Proc. IEEE/AIAA 40th Digital Avionics Syst. Conf. (DASC), 2021.

[24] H. Sakoe and S. Chiba, “Dynamic programming algorithm optimization for spoken word recognition,” IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 26, no. 1, pp. 43–49, 1978.

[25] L. Schwarz, “Estimating the dimension of a model,” Journal of the American Statistical Association, vol. 58, 1963.

[26] E. Eirola and A. Lendasse, “Gaussian Mixture Models for Time Series Modelling, Forecasting, and Interpolation,” International Symposium on Intelligent Data Analysis, Springer, 2013.

[27] G. Schwarz, “Estimating the dimension of a model,” The annals of statistics, 1978.

[28] W. Liu and I. Hwang, “Probabilistic trajectory prediction and conflict detection for air traffic control,” J. Guid., Control, Dyn., 2011.

[29] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, “Recurrent neural network based language model,” in Eleventh annual conference of the international speech communication association, 2010.

[30] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, no. 8, 1997.

[31] Y. Pang, H. Yao, J. Hu, and Y. Liu, “A recurrent neural network approach for aircraft trajectory prediction with weather features from sherlock,” in AIAA Aviation 2019 Forum, 2019.

[32] C.E. Seah and I. Hwang, “Stochastic linear hybrid systems: Modeling, estimation, and application in air traffic control,” IEEE Transactions on Control Systems Technology, no. 3, 2009.

[33] C. Gong and D. McNally, “A methodology for automated trajectory prediction analysis,” in Proc. AIAA Guid., Navig., Control Conf. Exhibit, Providence, RI, USA, Aug. 2004.

[34] W. Schuster, M. Porretta, and W. Ochieng, “High-accuracy four-dimensional trajectory prediction for civil aircraft,” Aeronaut. J., vol. 116, no. 1175, pp. 45–66, Jan. 2012.

[35] S. Ayhan and H. Samet, “Aircraft trajectory prediction made easy with predictive analytics,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.

HONG-CHEOL CHOI received the B.S. degree in applied physics from Korea Military Academy, Seoul, Korea, in 2012 and the M.S. degree in mechanical engineering from Seoul National University, Seoul, Korea, in 2016. He is currently pursuing the Ph.D. degree in aeronautics and astronautics from Purdue University, West Lafayette, IN.

His research interests include machine learning, data mining, and air traffic management.

CHUHAO DENG received the B.S. and the M.S. degree from Purdue university. He is currently pursuing the Ph.D. degree in aeronautics and astronautics from Purdue University, West Lafayette, IN. His research interests include air traffic management, data processing and machine learning.

INSEOK HWANG received the B.S. degree from Seoul National University, Seoul, Korea, and the M.S. degree from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, both in aerospace engineering, and the Ph.D. degree in aeronautics and astronautics from Stanford University, Stanford, CA, in 2004.

He is currently an Assistant Professor with the School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN. He also holds an affiliate appointment with the system-of-systems signature area at Purdue University. His research interests include control and information inference of complex networked embedded systems such as transportation systems, networked robotics, communication, and sensor networks, and biological systems, and their application to multiple-vehicle systems, especially to air traffic surveillance and control. For his research, he leads the Flight Dynamics and Control/Hybrid Systems Laboratory, Purdue University.

Dr. Hwang was a recipient of the NSF CAREER Award on stochastic hybrid systems and its application to mobile networked embedded systems in 2008. He is currently an associate fellow of the AIAA, and a member of the IEEE Control Systems Society and Aerospace and Electronics Systems Society.