IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 24, 2019, accepted August 4, 2019, date of publication August 9, 2019, date of current version August 22, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2934188

Abnormal Behavior Detection Scheme of UAV Using Recurrent Neural Networks

KE XIAO[1,2], JIANYU ZHAO[1], YUNHUA HE[1], CHAOFEI LI[1], AND WEI CHENG[2]
[1]School of Information Science and Technology, North China University of Technology, Beijing 100144, China
[2]School of Engineering and Technology, University of Washington, Tacoma, WA 98402, USA

Corresponding author: Yunhua He (heyunhua@ncut.edu.cn)

ABSTRACT With the development of technology and the decreasing of manufacturing costs, unmanned aerial vehicle (UAV) is considered to be one of the most effective relay to expand the communication coverage and improve the performance of cellular networks. However, the communication system of UAV is very susceptible to Global Positioning System (GPS) spoofing, causing it to deviate from the original trajectory and perform abnormal behavior. To address this issue, the abnormal behavior detection scheme of UAV using Recurrent Neural Networks (RNNs) is proposed in this paper. Specifically, the reliable normal behavior models for two different scenarios are established by applying RNNs to avoid the confusion of slight offset and abnormal behavior, so as to improve the accuracy of proposed detection scheme of UAV. Besides, in order to ensure the accuracy of training samples of RNNs, Direction of Arrival (DOA) estimation algorithm is used to obtain a large number of current 2D arrival angle of UAV. Moreover, an appropriate threshold is selected through amounts of experiments to measure the Normalized Root Mean Square Error (NRMSE) between the real position and the position provided by normal behavior models, thus detecting the abnormal behavior of UAV. Experimental results reveal that the proposed abnormal behavior detection scheme is of high accuracy.

INDEX TERMS UAV, abnormal behavior detection scheme, RNNs, GPS spoofing.

I. INTRODUCTION

Facing with the rapidly growing demand for high transmission rate and high communication coverage for wireless communication services, unmanned aerial vehicle (UAV) communication has recently become an active research area [1]. Since UAV can be deployed quickly in the air because of the specific maneuverability, it can not only provide the wireless service for some hotspots, but also offer signals to regional users instead of base station (BS) when terrestrial BS fails [2]. In addition, the flexible location of UAV can supply additional performance compared with fixed infrastructure based communications. Therefore, UAV is widely used as a communication relay to extend the communication range, thereby improving the performance of cellular network and satellite communication system [3].

However, these advantages of UAV also suffer from some challenges. UAV needs a reliable navigation system during the air communication, and the most common method to ensure the navigation is to build a state estimator around the sensor core, which is consisting of an inertial measurement unit (IMU) and a Global Positioning System (GPS) receiver. But the GPS receiver relies on weak satellite signals from about 20,000 kilometers from space, making the communication system very susceptible to the signal spoofing, which phenomenon is called GPS spoofing [4]. GPS spoofing is the behavior of producing the false GPS signal. Specifically, the purpose of controlling and guiding the location of UAV can be achieved by sending false location information to the GPS receiver [5]. When UAV is attacked by GPS spoofing, it will deviate from original trajectory and cause abnormal behavior.

Recently, solutions to GPS spoofing have also been extensively studied. References [6] and [7] proposed an architecture that can efficiently combined the visual feature information from a monocular camera with measurements from inertial sensors. Besides, the inertial measurements were used to predict frame-to-frame transition of online selected feature locations, and the difference between predicted and observed feature locations was used to account for initial misalignment

The associate editor coordinating the review of this article and approving it for publication was Shui Yu.

errors. However, the visual aided navigation required the preparation of maps in advance and it was limited by the environment, such as in the dark or misty.

Considering these defects, [8] proposed a non-cryptographic Global Navigation Satellite Systems (GNSS) anti-spoofing technique that ''sandwiches'' a spoofer between a correlation function distortion monitor and a total in-band power monitor. Reference [9] considered the consistency within GPS signals, and explored the relationship between the mean carrier to noise ratio value and the angular position of corresponding satellite. Specifically, the authors mentioned that this kind of relationship was quite stable for fixed receiver, and formed a spatial pattern. The uniqueness and complexity of this pattern qualified itself as an adequate fingerprint of the authentic signal, therefore it had the potential for discriminating spoofing signal. Besides, a method based on artificial neural network was proposed, which could detect the abnormity in this spatial pattern. However, the calculation process of these two methods is too cumbersome, the predecessor began to make some tests on the antenna software and compared multipath signatures to detect spoofing attack.

Since the GNSS were vulnerable to different sources of interference, especially affected by GPS spoofing, a Chisquare Goodness of Fit (GoF) based spoofing detection method was proposed in [10], and the Chi-square GoF test was applied at the correlators output of a software receiver with the purpose of detecting the presence of a spoofing attack; [11] compared the multipath signature of received signal, and the delay and gain ratio between multiple paths of wireless channels that cannot be masked by spoofing signals were exploited, so that the ratios of authentic satellite signals that experience independent channels would be independent of each other, GPS spoofing can therefore be detected.

Although the methods we mentioned above can achieve the detection of abnormal behavior, the ''difference calculation'' and ''relationship calculation'' in [6]–[9] is relatively complicated, and [10], [11] consume time to make tests on antenna software and make comparisons to multipath signatures. Therefore, it is necessary to propose a new scheme to realize real-time detection of anomalous behavior with high accuracy.

In recent years, deep learning has been widely used in many fields to train data models to achieve the acquisition and prediction of data with time sequence through the use of models. Reference [12] proposed a deep-learning based framework for reconstructing the missing data to facilitate analysis with remote sensing time series. Besides, this idea was to train the available data from both earlier and subsequent times-tamps, thus realizing the data prediction using the earlier part of the sequence. Reference [13] developed a novel integrated machine learning and coordinated beamforming solution to enable highly-mobile mmWave applications. Specifically, the developed solution leveraged a deep learning model to learn how to use the signatures of received signals, thus realizing the prediction of beamforming vectors at the BSs. Reference [14] considered to use deep learning network to predict the displacement between neighboring frames for each pedestrian, thus achieving the trajectory prediction of pedestrians. Inspired by above references, an abnormal behavior detection scheme of UAV using deep learning algorithm is considered, and a neural network with good memory is necessary to be applied to adapt to the characteristics of the angle data in UAV, so Recurrent Neural Networks (RNNs) are chosen here.

In this paper, the abnormal behavior detection scheme of UAV using RNNs is proposed. Since UAV is easily affected by wind and airflow during the air communication, a slight offset will occur on the original trajectory. In order to avoid the confusion of this slight offset and its abnormal behavior to improve the accuracy of the proposed abnormal behavior detection scheme, a reliable normal behavior model of UAV is needed. First, RNNs are applied to train and build the normal behavior model of UAV. In particular, Direction of Arrival (DOA) estimation algorithm is used to obtain a large number of current two-dimensional (2D) arrival angle of UAV as the training data of RNNs, thereby ensuring the accuracy of the position angle. Next, an appropriate threshold (which will be discussed in detail later) is selected through amounts of experiments. When Normalized Root Mean Square Error (NRMSE) between the real position and the position of trained normal behavior model of UAV is greater than the chosen threshold, it is detected that the UAV has abnormal behavior, thus achieving the purpose of abnormal behavior detection of UAV. The contributions are summarized as below.

(1) In order to avoid the confusion of UAV's slight offset and its abnormal behavior to improve the accuracy of the proposed abnormal behavior detection scheme, RNNs are applied to train and establish a reliable normal behavior model of UAV;

(2) In order to demonstrate the applicability of the normal behavior model trained in this paper, it is applied to two scenarios, which are smart city and highway respectively. Besides, the optimal parameters of the model in two scenarios are obtained by experiments.

(3) Experimental results reveal that the accuracy of proposed detection scheme in these two scenarios is about 98% in average and the real-time detection of abnormal behavior is realized by applying the trained ready normal behavior model. Therefore, the security of the entire communication system can be improved.

The remainder of this paper is organized as follows. The detailed problem description is provided in Section II. In Section III, we thoroughly describe the establishment process of RNNs-based normal behavior model. Extensive experimental results and analysis are presented in Section IV. Finally, Section V concludes the paper.

II. PROBLEM DESCRIPTION

Since UAV is easy to operate and deploy, and it can effectively improve the coverage area of entire communication system,

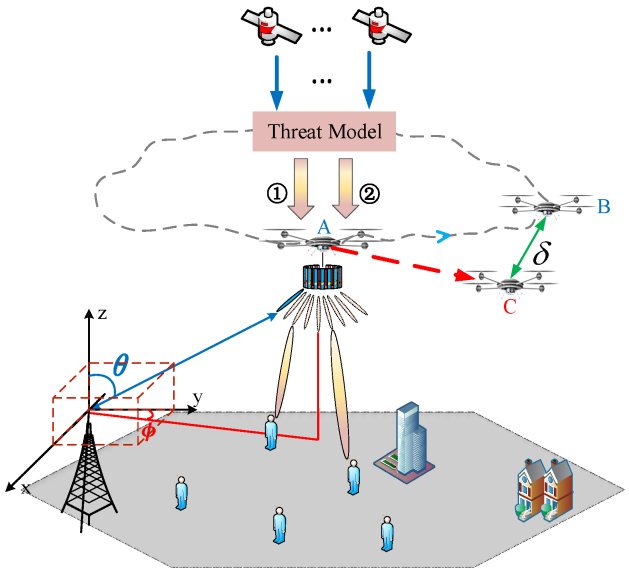


FIGURE 1. Communication scenario of UAV attacked by GPS spoofing.

it is widely used as a communication relay in various communication scenario. However, the UAV is vulnerable to GPS spoofing and deviates from original trajectory in the process of air communication, and the specific communication scenario of UAV attacked by GPS spoofing is shown in Figure 1.

As can be seen from the figure, UAV is flying as a communication relay above the scene to provide reliable signal transmission for all users. In order to improve the signal coverage, the cylindrical antenna array is applied to achieve 360-degree beamforming, and the angles at which UAV floats up and down, left and right are called elevation angle θ and horizontal angle ϕ, respectively. Besides, GNSS are needed to continuously guide the location during the flight of UAV. When UAV is attacked by GPS spoofing, it will perform abnormal behavior and deviate from the position of normal behavior model "B" to the abnormal position "C", and the model composed of this attack method can be called the "threat model" in the figure.

It is well known that GPS positioning adopts the principle of Three-sphere positioning principle. That is, after the GPS signal propagation delay f is obtained by the time difference measurement, the pseudo-distance r_0 between the satellite and the receiver ($r_0 = Cf$, C is the propagation speed of the light) can be calculated. As long as the pseudo-distance of the positioning receiver to three satellites is measured, three spherically connected equations in space can be obtained, and the (x, y, z) coordinates of the receiver can be obtained by calculation. But in fact, there is always a clock difference between the receiver and the satellites' standard clock. Therefore, the pseudo-distance of the fourth satellite must be measured, and the equations of the first three spherical equations are connected to calculate the timing error [15].

Let the position "A" of the known UAV be (x_A, y_A, z_A), and the pseudo-distances corresponding to the four satellites forwarded are (d_1, d_2, d_3, d_4), and the timing error of the UAV is f_A, then the positional equation of point "A" is,

$$\begin{cases} d_1 = [(x_A - x_1)^2 + (y_A - y_1)^2 + (z_A - z_1)^2]^{\frac{1}{2}} + Cf_A, \\ d_2 = [(x_A - x_2)^2 + (y_A - y_2)^2 + (z_A - z_2)^2]^{\frac{1}{2}} + Cf_A, \\ d_3 = [(x_A - x_3)^2 + (y_A - y_3)^2 + (z_A - z_3)^2]^{\frac{1}{2}} + Cf_A, \\ d_4 = [(x_A - x_4)^2 + (y_A - y_4)^2 + (z_A - z_4)^2]^{\frac{1}{2}} + Cf_A, \end{cases} \quad (1)$$

where the coordinates of the four satellites are (x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3) and (x_4, y_4, z_4), respectively, f_A is the specific GPS signal propagation delay of the known UAV that can be obtained by time difference measurement.

After the attacker locks the attacked object (GPS receiver), the attacker can generally perform spoofing attacks in two ways(as can be seen in threat model of the figure): one is to calculate the pseudo-distance d by using GPS ranging, and the high-fidelity processing is performed on the received satellite signals by a jammer, and then delayed and forwarded, misleading the GPS receiver to calculate the wrong pseudo-distance (larger than the real value), which is called a forward GPS spoofing attack. We can take the first satellite for example, when it is interfered, d_1 in the equation will be the wrong value, so the de-positioning equation necessarily obtains the wrong positioning information of the receiver, thereby achieving the purpose of disturbing and deceiving, and the wrong position information (x_A, y_A, z_A) of UAV will be obtained. The other is that the attacker will directly forges GPS interference signal according to the characteristics of the satellite signals that can be received in the area. Besides, the forged signal is broadcasted as a satellite signal to the signal receiving area where the attacked object is located, and the attacked GPS receiver is induced to lock the interference signal source to obtain the wrong pseudo-distance d and positioning coordinates (x_A, y_A, z_A) of UAV.

Both GPS spoofing methods we mentioned above have the same purpose of obtaining an erroneous pseudo-distance, so that the target receiver can generate a positioning error, deviate from the original trajectory and perform abnormal behavior. Therefore, an abnormal behavior detection scheme of UAV using RNNs is proposed to detect abnormal behavior in time. Specifically, a novel normal behavior model of UAV is trained through the current 2D arrival angle obtained by DOA estimation algorithm. According to the trained normal behavior model, the abnormal behavior of UAV can be measured and detected based on the NRMSE between abnormal position "C" and the position point "B" of normal behavior model(gray curve in Figure 1). When NRMSE is larger than the chosen threshold δ(through experiments), the behavior can be detected as abnormal; but when NRMSE is less than δ, the behavior will be considered to be a normal behavior of a slight offset by factors such as wind and airflow.

III. RNNS-BASED NORMAL BEHAVIOR MODEL OF UAV

Since whether the established normal behavior model of UAV is reliable or not will directly affect the performance of proposed abnormal behavior detection scheme, the

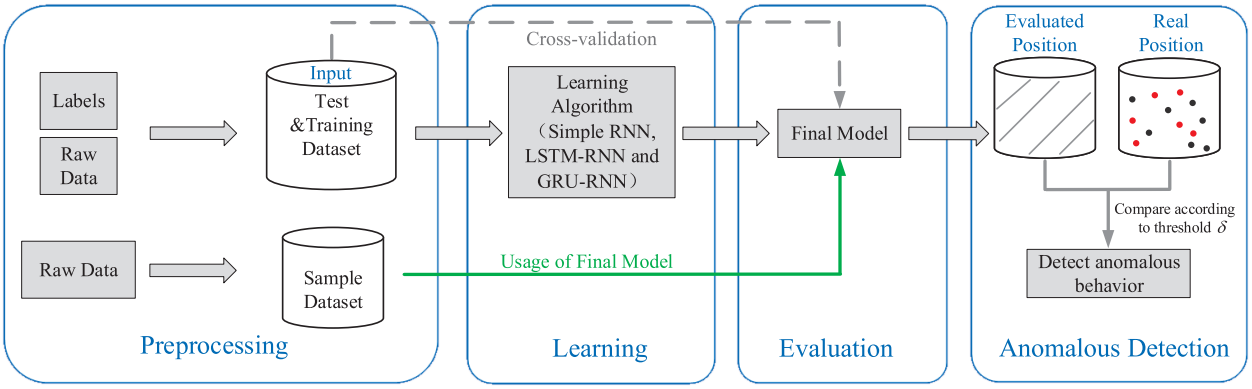


FIGURE 2. Establishment process of RNNs-based normal behavior model(Black points represent normal angle data, and red points represent abnormal angle data).

specific establishment process of RNNs-based normal behavior model is introduced in detail in this section, which can be seen in Figure 2.

First, the 2D arrival angle (elevation angle θ and horizontal angle ϕ) of UAV is obtained based on the DOA estimation algorithm of cylindrical antenna array. In particular, we integrate the elevation and horizontal angles into a one-dimensional (1D) angle to improve the convergence speed of program fitting during preprocessing of data. In our program, the step size is designed as 20, so the raw data is consist of 20 angle data, and the label is the 21th data, both of which can form the training dataset, and serve as an input to train the model. Next, we input processed training dataset into three different network models including Simple RNN, Long Short Term Memory based RNN (LSTM-RNN) and Gated Recurrent Unit based RNN (GRU-RNN) for learning. By comparing the reliability of three behavior models, the best RNNs-based normal behavior model is chosen, which is called final model in the figure. In addition, it is worth mentioning that the cross-validation can be realized while the model is being trained to evaluate the reliability of the model. Finally, the sample dataset acquired from a given trajectory will be input into the trained final model to obtain the evaluated position, thereby realizing the use of final model. And then, the NRMSE between obtained evaluated position and real position is calculated. If the NRMSE between them is greater than threshold δ, the behavior of the UAV is considered abnormal at that moment. Therefore, the abnormal behavior detection of UAV can be achieved.

A. ACQUISITION OF 2D ARRIVAL ANGLE OF DOA ESTIMATION

The current 2D arrival angle (elevation angle θ and horizontal angle ϕ in Figure 1) can be calculated by using DOA estimation algorithm of cylindrical antenna array, so it is necessary to investigate the DOA estimation algorithm formula for cylindrical antenna array.

Suppose that there are M array elements uniformly distributed on the circumference, and if the center of the circle is the coordinate origin, the coordinates of the mth array element are (r, ϕ_m), where r is the radius of cylinder and ϕ_m is the horizontal angle of the array element, $\phi_m = 2(m-1)\pi/M$, $(m = 1, 2, \ldots, M)$ [16]. If it is assumed that $k (k < M)$ uncorrelated intra-frequency narrow-band signals are incident on the array from the far-field $\theta_i (i = 1, 2, \ldots, k)$ direction, and the signals are in the same plane as the array, the data received by the array at time t can be expressed as [17],

$$\mathbf{X}(t) = \mathbf{F} \cdot \mathbf{A}\mathbf{S}(t) + \mathbf{N}(t) \tag{2}$$

where $\mathbf{X}(t) = [\mathbf{x}_1(t), \mathbf{x}_2(t), \ldots, \mathbf{x}_M(t)]^T$ is the received signal vector. $\mathbf{F} = [\mathbf{f}(\theta_1), \mathbf{f}(\theta_2), \ldots, \mathbf{f}(\theta_k)]^T$ is the radiation plot matrix of array elements, and $\mathbf{f}(\theta) = [\mathbf{f}(\theta - \phi_1), \mathbf{f}(\theta - \phi_2), \ldots, \mathbf{f}(\theta - \phi_M)]^T$, $\mathbf{f}(\theta - \phi_M)$ is the radiation plot of a single element. When $\mathbf{f}(\theta - \phi_M) = 1$, it means that the array elements are omnidirectional elements [18]. $\mathbf{S}(t) = [\mathbf{s}_1(t), \mathbf{s}_2(t), \ldots, \mathbf{s}_k(t)]^T$ is the complex amplitude vector of each signal, and $\mathbf{N}(t) = [\mathbf{n}_1(t), \mathbf{n}_2(t), \ldots, \mathbf{n}_M(t)]^T$ is the white noise vector. $\mathbf{A} = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), \ldots, \mathbf{a}(\theta_k)]$ is the array manifold matrix, where $\mathbf{a}(\theta_i)$ is the director of ith signal, and $\mathbf{a}(\theta_i) = [e^{j\frac{2\pi r}{\lambda}cos(\theta_i-\phi_1)}, e^{j\frac{2\pi r}{\lambda}cos(\theta_i-\phi_2)}, \ldots, e^{j\frac{2\pi r}{\lambda}cos(\theta_i-\phi_M)}]^T$, λ is the wavelength of the signal.

Different from the general circular array, cylindrical antenna array is composed of the mental cylinder, which is also the reason of "shadow effect" phenomenon. That is, only some of the array elements of cylindrical antenna can receive the signal that incident in a certain direction. Take a 16-element array as an example, when a signal is incident on the array from the $\theta = \frac{\pi}{2}$, only the elements 1-9 can receive the signal [19]. The steering matrix $\mathbf{a}(\frac{\pi}{2})$ of this signal at this moment can be represented as $[1, e^{\frac{j2\pi}{\lambda}cos(\frac{\pi}{2}-\frac{\pi}{8})}, e^{\frac{j2\pi}{\lambda}cos(\frac{\pi}{2}-\frac{\pi}{4})}, \ldots, e^{\frac{j2\pi}{\lambda}cos(\frac{\pi}{2}-\pi)}, 0, \ldots, 0]^T$, where "0" means the obscured elements of the array, which will cause the performance of the algorithm to deteriorate dramatically.

In order to eliminate the incomplete signal steering vector caused by the "shadow effect" of mental cylinder, a method to divide the entire array elements into several sub-arrays is needed. The principle of division is that for an incident

signal, at least one sub-array should be guaranteed, and all the elements of the sub-array can receive the signal. For cylindrical antenna array, it should be ensured that the span of each sub-array $\leq \frac{\pi}{2}$. Taking the 16-element array as an example, the array can be divided into 8 sub-arrays: array elements 1-5 are the first sub-array, and array elements 3-7 are the second sub-array, and so on. By using this method, a sub-array can always be found for any incident signal, and all its elements of the sub-array can receive the signal. Therefore, the steering vector of the signal on the sub-array is ''complete''. The specific steps are as follows [20]–[22],

- Step 1: Use Formula (2) to get the data vector of the whole array,

$$\mathbf{X}(t) = [\mathbf{x}_1(t), \mathbf{x}_2(t), \ldots, \mathbf{x}_M(t)]^T \tag{3}$$

- Step 2: Extract the data vector of each sub-array from the first step $\mathbf{X}_c(t) = [\mathbf{x}_{c1}(t), \mathbf{x}_{c2}(t), \ldots, \mathbf{x}_{cs}(t)]^T$, where c represents the cth sub-array, and s stands for the number of array elements in the sub-array.
- Step 3: Constructing a spatial spectrum calculation formula for the cth sub-array using the MUSIC algorithm.

$$P_c = \frac{1}{\mathbf{a}_c^H(\theta)\mathbf{U}_c\mathbf{U}_c^H\mathbf{a}_c(\theta)}, \theta \in \theta_c \tag{4}$$

where $\mathbf{a}_c(\theta)$ represents the steering vector of the signal on the cth sub-array. $\mathbf{U}_c$ is the $s \times (s - p_s)$ dimensional matrix, whose column vector consists of the feature vector of the sub-array's noise sub-space. $p_s (p_s < s)$ is the number of signals incident on the sub-array c, and θ_c is the search range of the cth sub-array.

Assume that there is a 3×8 cylindrical antenna array, and the radius $r = 0.6\lambda$ and the height $H = 2\lambda$. Besides, the distance between adjacent array elements is defined as about 0.5λ. According to the three-step principle of sub-array division, a total of 9 array elements per three columns are divided into one sub-array, and the entire array is divided into 8 sub-arrays. From the symmetry, 8 array elements in each layer have the same ϕ angle and different θ angels of three elements in each column, but the top and bottom array elements are symmetrical. Suppose that one source signal is incident on the array from $(\phi, \theta) = (60, 50)$ direction, and the SNR is 10dB, the number of samples is 100. According to the calculation formula (4) of spatial spectrum, the simulation result of the DOA estimation algorithm can be obtained, which is shown in Figure. 3. According to the highest peak, the elevation angle θ and horizontal angle ϕ can be estimated accurately.

B. DATA PREPROCESSING OF RNNS-BASED NORMAL BEHAVIOR MODELS

In order to improve the quality and training efficiency of RNNs-based normal behavior models, a series of complex data preprocessing is required for the acquired raw data [23], and the specific process of data preprocessing is shown in Figure. 4.

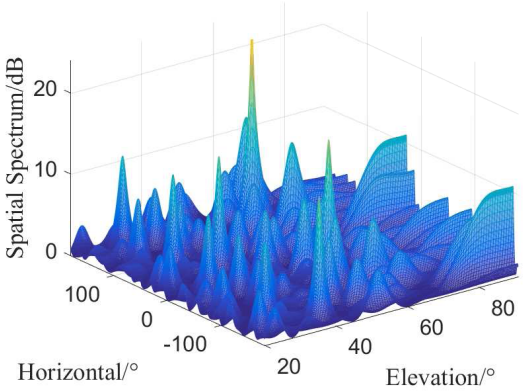


FIGURE 3. Spectrum of DOA estimation algorithm.

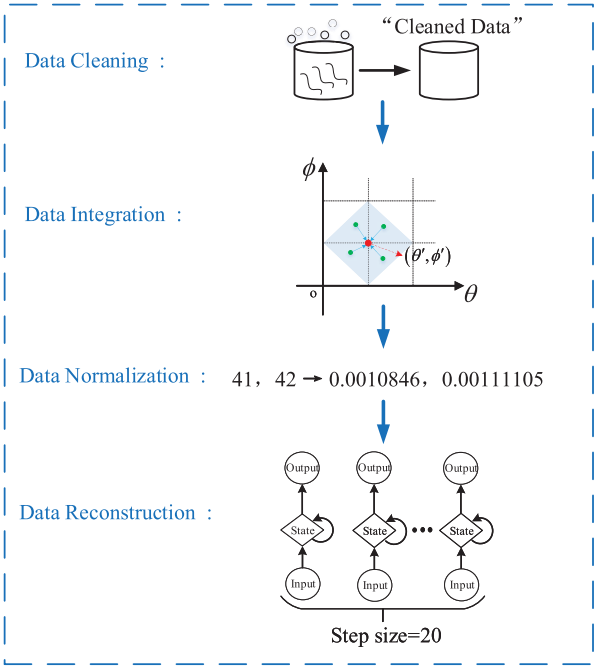


FIGURE 4. Specific process of data preprocessing.

First of all, data cleaning is needed for acquired angle data. Data cleaning can effectively detect and correct identifiable errors in data files, including checking the consistency of data, processing invalid values and missing values, etc., thus completing the data verification, and obtaining ''cleaned data''. In this paper, data cleaning is focused on clearing the data that performs empty in the collected data, and filling the null data by using the sliding window to take the mean value.

Besides, if two 1D arrival angles (elevation angle θ and horizontal angle ϕ) are input into RNNs model separately, the training process will be too cumbersome and the results will not be accurate. Therefore, data integration is performed next. Specifically, the elevation angle is supposed as x-axis, horizontal angle as y-axis, and the unit length is set to 0.1 to reduce the error during the integration process. In the established coordinate system, each point is a set of 2D coordinates

at a certain moment, and when the coordinates are not on the axis of the unit length, we do decimals to round up and round down the coordinates. As shown in Figure. 4, all points in the blue area will be regarded as the red point (θ', ϕ'), and then, this coordinate point can be transferred into a unique value z according to a function $z = f(\theta', \phi')$, thereby achieving the data integration.

In addition, in order to adapt to the activation function of RNNs, the integrated angle data will be mapped to the 0-1 range, thus speeding up the convergence speed of the program.

Finally, the discrete data that collected by time node will be reconstructed into the input data with a time step size of 20 to improve the memory of the RNNs model, thus improving the accuracy of the normal behavior model of UAV. Finally, the data is saved to the local disk for future use.

C. ESTABLISHMENT AND APPLICATION OF RNNS-BASED NORMAL BEHAVIOR MODELS

Since the trained 2D arrival angle obtained by DOA estimation is in the form of time sequence, RNNs are used to train the normal behavior model of UAV.

As a basic model of deep learning, Simple RNN solves the problem of memorylessness in traditional neural networks. Suppose that h_{t-1} and h_t represent the hidden state of time step $t-1$ and t, x_t and O_t are the states of input data and output data at time step t. W_* and b_* stand for the weight vectors and bias vectors, respectively [24]. Therefore, the relationship between an input sequence to an output sequence is a mapping function, which can be specified by,

$$h_t = tanh(W_h[x_t, h_{t-1}] + b_h) \tag{5}$$
$$O_t = W_O h_t + b_O \tag{6}$$

Specifically, *tanh* function in equation (5) is a hyperbolic tangent function and the output is limited between -1 and 1. When the function independent variable tends to be positive infinity or negative infinity, the function approaches a smooth state, and it is mainly used in cell state C, hidden state h and the candidate state $\tilde{h}_t$. Since these states are the states information after screening, which are beneficial to the input and processing for the next moment, the use of *tanh* in the loop process can continuously expand the feature effect and deepen the influence of current moment on the next moment, thereby make the transmission of information have a better effect.

Besides, LSTM-RNN can store the information for longer period of time with the help of Keep Gate or Forget Gate, and "Gate" controls the cell state of LSTM and decides to remove or add the information to the neural network [25]. The relationship between an input to an output sequence can be specified by,

$$i_t = \sigma(W_i[x_t, h_{t-1}] + b_i) \tag{7}$$
$$f_t = \sigma(W_f[x_t, h_{t-1}] + b_f) \tag{8}$$
$$o_t = \sigma(W_o[x_t, h_{t-1}] + b_o) \tag{9}$$
$$C_t = f_t C_{t-1} + i_t tanh(W_c[x_t, h_{t-1}] + b_c) \tag{10}$$
$$h_t = o_t tanh(C_t) \tag{11}$$

where i, f, o and c are respectively the input gate, forget gate, output gate and cell state, σ represents the logistic sigmoid function. It is worth mentioning that the sigmoid function is typically used for hidden layer neuron output, which maps a real number to the interval of (0,1). At the same time, when the function independent variable tends to positive infinity or negative infinity, the function approaches the smooth state. When the independent variable tends to positive infinity, the output is infinitely close to 1, and when the independent variable tends to be infinite, the output is infinitely close to 0. Therefore, the sigmoid function is mainly applied to the output of the gate structure such as LSTM-RNN and GRU-RNN, and the output result is divided into 0 and 1, so that the state information is set in the process of training iteration (setting the state information unfavorable to the next moment to 0, setting the state information that is beneficial to the next moment to 1) to ensure that each iteration can filter the useless state information and deliver useful state information.

While GRU-RNN retains LSTM-RNN's resistance to the vanishing problem, but its internal structure is simpler [26]. Therefore, GRU-RNN is easier to train since fewer calculations are needed to update its hidden state, and the forward propagation process of GRU-RNN can be expressed as [27],

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{12}$$
$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \tag{13}$$
$$\tilde{h}_t = tanh(W_h \cdot [r_t * h_{t-1}, x_t]) \tag{14}$$
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \tag{15}$$

where r_t and z_t represent the reset gate and update state, respectively, $\tilde{h}_t$ denotes the candidate state.

Assume that RNNs are applied to the field of natural language processing, each data sample can be considered to be a time series, and there is a certain correlation between each data sample and data of previous and next moment. Therefore, analyzing the relationship between the data samples requires the hidden layer unit (h_t) in the RNNs. The hidden neurons are used as the output of the data samples at the previous moment and the input of the data samples at next moment, ensuring the continuity and sequence of the statements. As mentioned above, there is only a single use of hidden layer neurons (h_t) in Simple RNN, but gradient explosions and gradients are prone to occur when sentences are too long; and the gated structure proposed by LSTM-RNN (forgotten gate, input gate and output gate) solves the above two problems, thereby ensuring that the vocabulary information in the longer sentences can be passed to the back to improve the semantic relevance; GRU-RNN combines the forget gate and input gate of LSTM-RNN into a single update gate, and integrates the cell state and hidden state of the data sample, so it can not only learn the long-term dependence

of the text, but also has a shorter processing time than LSTM-GRU.

Similarly, this paper applies RNNs to train UAV angle data with time series, which can also achieve the same effect. Besides, although GRU-RNN has the best performance in the training process, the memory capacity of the required neural network training model is diverse for different kinds of training data, which means that GRU-RNN with the best memory capacity does not necessarily make the final model have the best performance. Therefore, the reliability of three RNNs-based normal behavior models needs to be experimented and compared to obtain the best final model. After comparing the reliability of three models, the structures of best normal behavior models that adapted to the UAV communication scenario of this paper are selected to realize the acquisition and establishment of the normal behavior models of UAV.

Specifically, in the process of using normal behavior model, some abnormal behavior data of UAV is interspersed with other normal behavior data. In addition, the same data preprocessing method is applied to reconstruct the 2D arrival angle of abnormal behavior of UAV to adapt to the structures of established normal behavior models. By inputting the angle data (including normal behavior and abnormal behavior) into the normal behavior models for training, the data under the normal behavior models can be obtained. Next, the NRMSE between the trained position (which can be obtained from 2D arrival angle) and real position can be calculated. If NRMSE is greater than the threshold δ, which can be obtained through amounts of experiments, the coordinate point is recorded as abnormal data, otherwise it is recorded as normal data. Therefore, the abnormal behavior of UAV can be detected by comparing the abnormal labels of the training dataset with the abnormal points detected by the normal behavior models.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section,the experimental results of proposed RNNs-based normal behavior models are presented and analyzed through Python, including Simple RNN, LSTM-RNN and GRU-RNN respectively to establish the most reliable normal behavior models of UAV. Besides, a Python-based sequence processing tool Tensorflow is applied to build the deep learning models and obtain the parameters of the optimal model.

Besides, it is hard to obtain the angle data of UAV while it is flying in the air because of the limitations of existing experimental conditions. Therefore, this paper establishes a stable simulation platform by analyzing the real trajectory of Dajiang mavic Pro UAV and combining the national standards provided in the ITU-TY.IoT-UAS-Reqts (Use cases, requirements and capabilities of unmanned aircraft systems for Internet of Things) recommendations, which is supported by International Telecommunication Union (ITU). Specifically, it is worth mentioning that in the process of setting up the simulation platform, the volatility and instability of the wind are considered, and proper signal-to-noise ratio (SNR) is added to x, y and z axis directions of the UAV's position information, respectively. Besides, some strong winds with higher SNRs are randomly interspersed, which is more in line with the wind's volatility. The SNR values of gentle wind in x, y and z axis are 50, 50 and 25, and the values of strong wind are 100, 100 and 50 respectively. Therefore, the $x(t)$ in formula (2) is reliable, thus acquiring the accurate 2D arrival angle by using DOA estimation algorithm. In addition, the training epoch of RNNs is defined as 500 to ensure the convergence of accuracy curves.

In addition, the proposed normal behavior model is applied to two communication scenarios to prove the applicability. Since the signal transmission provided by BS cannot cover every user in smart city, UAV is introduced as a communication relay to expand the coverage area. Moreover, when the highway is located in the blind spot area of communication between two BSs, UAV can also provide the signal transmission for vehicles and pedestrians. Therefore, the normal behavior model proposed in this paper is applied to above two typical UAV communication scenarios, respectively. Specifically, the optimal parameters of normal behavior model can be selected by implementing the experimental results, so as to obtain the optimal normal behavior model in these two scenarios.

In order to make sure the convergence speed of program running and the accuracy of proposed normal behavior model, the cross-validation is applied in the process of building normal behavior model. Specifically, the total number of data samples is 30,000, the first 20,000 data samples are used as the train set to train the RNNs-based models, and the remaining 10,000 data samples are applied as the test set to ensure the accuracy of the model training, and the step size of each sample is designed as 20. Next, the experimental results of three RNNs-based normal behavior models in two different scenarios will be presented and analyzed, respectively.

A. TRAINING OF NORMAL BEHAVIORAL MODELS FOR TWO SCENARIOS

1) NORMAL BEHAVIOR MODELS OF UAV IN SMART CITY SCENARIO

The experimental results of three different neural networks (including Simple RNN, LSTM-RNN and GRU-RNN) in smart city scenario are first presented in this part to obtain the most reliable final model of smart city scenario.

a: SIMPLE RNN

In order to select the optimal parameters of Simple RNN, a series of experiments are done by changing the number of layers and the number of neurons, and the experimental results of different network structures of Simple RNN in terms of Mean Square Error (MSE) are shown in Table 1. It can be seen from the table that when the number of neurons is less than 32, MSEs of all network structures including NET1-NET6 are relatively small and stable, resulting in a higher reliable probability of the normal behavior model of UAV. Besides, when the number of neurons is 16 and 32,

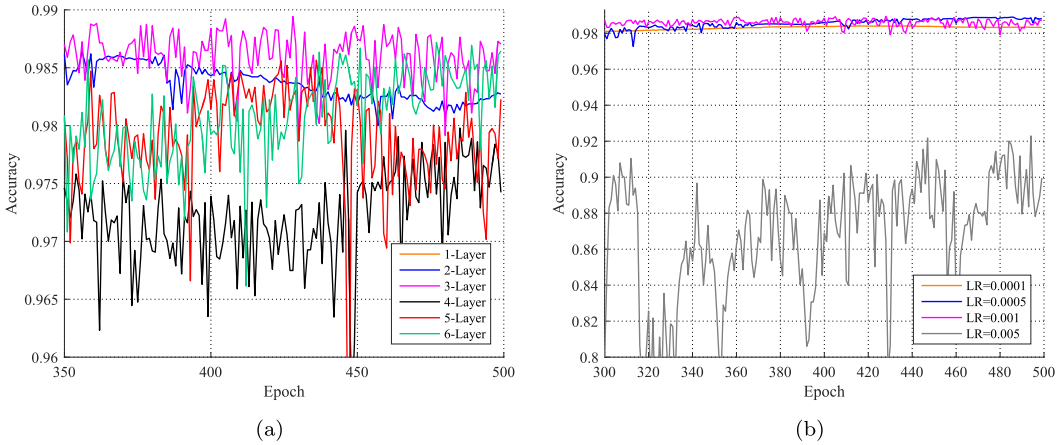


FIGURE 5. Accuracy of Simple RNN under different layers and LRs of smart city scenario. (a) Accuracy different layers. (b) Accuracy under different LRs.

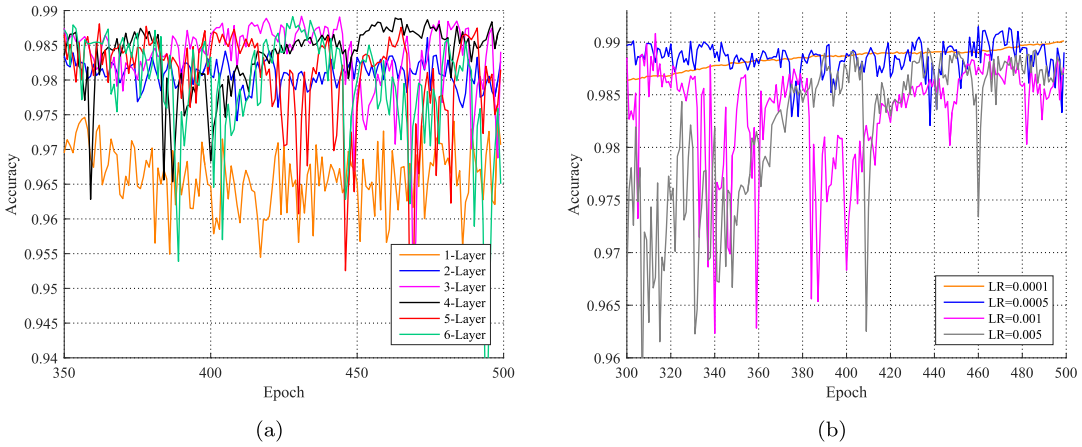


FIGURE 6. Accuracy of LSTM-RNN under different layers and LRs of smart city scenario. (a) Accuracy under different layers. (b) Accuracy under different LRs.

TABLE 1. MSE of different Simple RNN structures of smart city scenario.

Name	Structure	Test Set (Average MSE)					
		8	16	32	64	96	128
NET1	1-layer	0.0128	0.0107	0.0101	0.0095	0.0109	0.0106
NET2	2-layer	0.0082	0.0083	0.0100	0.0108	0.0116	0.0115
NET3	3-layer	0.0121	0.0066	0.0084	0.0429	0.0116	0.0119
NET4	4-layer	0.0067	0.0080	0.0102	0.0118	0.0136	0.0126
NET5	5-layer	0.0079	0.0075	0.0102	0.0552	0.0130	0.1100
NET6	6-layer	0.0074	0.0100	0.0101	0.0118	0.0140	0.0132

the value of MSE is the smallest and the fluctuation is the smoothest. Therefore, the number of neurons is designed as 16 to ensure the convergence speed of the program (more neurons, slower the program converges).

However, it is difficult to directly determine the optimal layers of Simple RNN according to the average MSE in Table 1, so the MSE data with 16 neurons in Table 1 is further converted into the reliable probability (which is called accuracy in the Figure) curve to more intuitively select the optimal layers, and the experimental results are shown in Figure 5(a), where the horizontal coordinate is training epoch and the vertical coordinate is accuracy, and the results show that 3-layer Simple RNN is the best choice to achieve the highest accuracy. Besides, Figure 5(b) presents the experimental results of different network structures with different learning rates (LRs) in terms of accuracy when the neurons and layers of all curves are defined as 16 and 3. As can be seen from Figure 6(b), Simple RNN always has the best performance when LR is equal to 0.0005. Therefore, when LR of the 3-layer Simple RNN with 16 neurons is equal to 0.0005, Simple RNN can achieve the best performance.

b: LSTM-RNN

Table 2 illustrates the MSE of different LSTM-RNN network structure by changing the number of network neurons and layers. It can be seen from the table that when the number of neurons is less than 32, the experimental results of MSE are relatively volatile as the number of layers changes, and MSE value is also relatively high. Besides, if the number

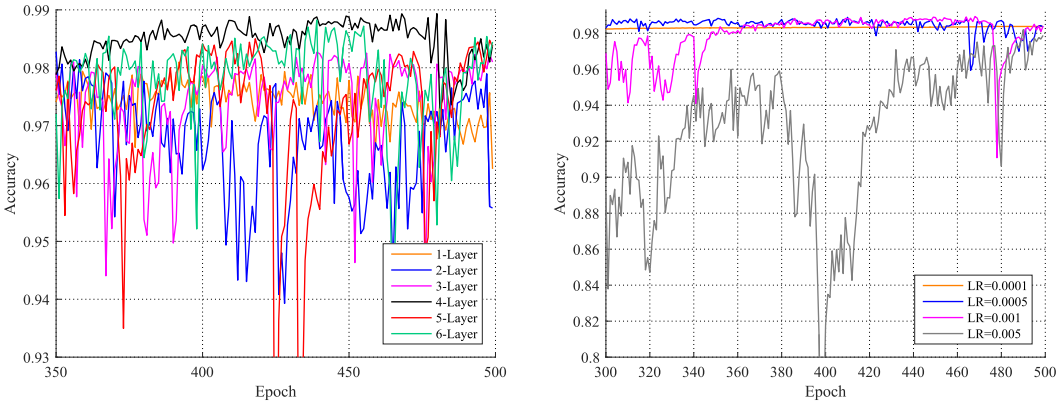


FIGURE 7. Accuracy of GRU-RNN under different layers and LRs of smart city scenario. (a) Accuracy under different layers. (b) Accuracy under different LRs.

TABLE 2. MSE of different LSTM-RNN structures of smart city scenario.

Name	Structure	Test Set (Average MSE)					
		8	16	32	64	96	128
NET1	1-layer	0.0086	0.0081	0.0089	0.0098	0.0099	0.0100
NET2	2-layer	0.0074	0.0084	0.0088	0.0097	0.0098	0.0099
NET3	3-layer	0.0093	0.0091	0.0095	0.0101	0.0103	0.0105
NET4	4-layer	0.0104	0.0096	0.0097	0.0104	0.0107	0.0110
NET5	5-layer	0.0102	0.0100	0.0102	0.0104	0.0113	0.0112
NET6	6-layer	0.0111	0.0106	0.0104	0.0110	0.0116	0.0117

TABLE 3. MSE of different GRU-RNN structures of smart city scenario.

Name	Structure	Test Set (Average MSE)					
		8	16	32	64	96	128
NET1	1-layer	0.0078	0.0077	0.0088	0.0090	0.0098	0.0104
NET2	2-layer	0.0084	0.0083	0.0090	0.0092	0.0097	0.0100
NET3	3-layer	0.0091	0.0095	0.0092	0.0097	0.0101	0.0106
NET4	4-layer	0.0080	0.0091	0.0097	0.0105	0.0103	0.0107
NET5	5-layer	0.0100	0.0098	0.0099	0.0103	0.0105	0.0111
NET6	6-layer	0.0100	0.0098	0.0102	0.0108	0.0107	0.0115

of neurons is too high, the training of the program will be too complicated, which will affect the convergence speed. Therefore, 32 neurons is used for LSTM-RNN model.

In order to better determine the optimal layers of LSTM-RNN model, the column with 32 neurons is converted into six accuracy curves, which is shown in Figure 6(a). It can be seen intuitively from the figure that 4-layer LSTM-RNN can achieve the highest average accuracy, and the performance of 4-layer LSTM-RNN is significantly more stable than that of other layers. In addition, Figure 6(b) shows the experimental results of 4-layer LSTM-RNN model with 32 neurons under different LRs. It can be seen from the figure that the accuracy results of 0.001 and 0.005 are relatively unstable, and when LR is 0.0005, the average accuracy result is higher than the most stable orange curve. Therefore, when LR of 4-layer LSTM-RNN with 32 neurons is 0.0005, the average highest accuracy 98.7% can be achieved to ensure the accuracy of the normal behavior model.

c: GRU-RNN

Table 3 shows the experimental results of MSE under different GRU-RNN network structures, when the number of neurons is not less than 32, a relatively low MSE value can be achieved, so 32 neurons is tend to be chosen to make sure the convergence speed. Besides, as the number of network layers changes, the MSE value changes smoothly. Therefore, the network model with 32 neurons is considered to have the lowest MSE.

Figure 7(a) illustrates the accuracy results of behavior model with 32 neurons in terms of different layers, which can more accurately show the performance of the model under different layers. As can be seen from the figure, 4-layer GRU-RNN model performs best in terms of accuracy, which is about 98.5%. Besides, Figure 7(b) shows the accuracy of 4-layer GRU-RNN with 32 neurons under different LRs. As can be seen from the figure, the curve of 0.005 LR is always in an unconverged state. And when LR is 0.0005, the average accuracy is better than other two curves, which is around 98%. Therefore, the conclusion can be made that when LR of 4-layer GRU-RNN with 32 neurons is 0.0005, the best accuracy can be achieved.

In order to accurately obtain the optimal normal behavior model of UAV under smart city scenario we discussed before, the accuracy of these three RNNs is concluded and compared, and the comparison results are shown in Figure 8.

As shown in Figure 8, the accuracy of normal behavior models under three different RNNs is compared in smart city scenario. It can be seen that all three accuracy curves remain rising, and when the number of training epoch is less than 150, the rising speed of LSTM-RNN is relatively fast, while the rising speed of LSTM-RNN and GRU-RNN is relatively slow. However, when the number of epoch reaches 450,

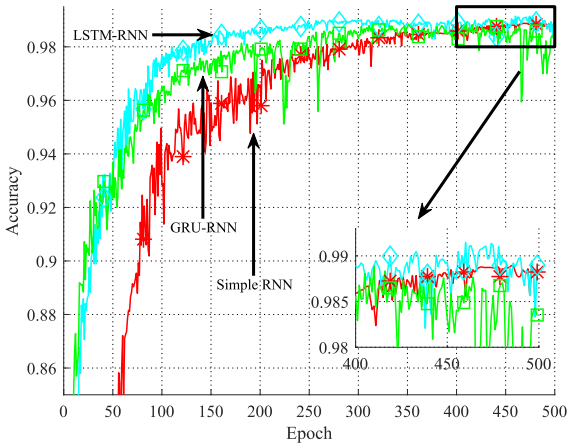


FIGURE 8. Accuracy comparison under three different RNNs in smart city scenario.

the accuracy curves of Simple RNN and LSTM-RNN have exceeded that of GRU-RNN (which can be clearly seen from the enlarged view), and the average accuracy curve of LSTM-RNN is higher than that of Simple RNN, so LSTM-RNN is applied as the optimal normal behavior model of smart city scenario.

2) NORMAL BEHAVIOR MODELS OF UAV IN HIGHWAY SCENARIO

In the highway scenario, the same idea as smart city scenario is used to select the optimal normal behavior model in terms of Simple RNN, LSTM-RNN and GRU-RNN. Through a series of experimental tests and comparisons, the Simple RNN model is chosed as the optimal normal behavior model of highway scenario with following experimental results.

TABLE 4. MSE of different Simple RNN structures of highway scenario.

Name	Structure	Test Set (Average MSE)					
		8	16	32	64	96	128
NET1	1-layer	0.0056	0.0036	0.0040	0.0036	0.0038	0.0035
NET2	2-layer	0.0065	0.0041	0.0034	0.0038	0.0037	0.0038
NET3	3-layer	0.0065	0.0036	0.0037	0.0047	0.0045	0.0052
NET4	4-layer	0.0063	0.0031	0.0043	0.0049	0.0058	0.0067
NET5	5-layer	0.0121	0.0038	0.0051	0.0059	0.0070	0.0093
NET6	6-layer	0.0066	0.0051	0.0054	0.0063	0.0075	0.0097

Table 4 shows the MSE values under different layers and different neurons of Simple RNN in highway scenario. As can be seen from the table, the lowest MSE can be achieved when Simple RNN is with 16 neurons, which will not be affected by the number of network layers. Therefore, the conclusion can be made from the table that Simple RNN with 16 neurons performs best in terms of MSE value. Besides, the accuracy of Simple RNN with 16 neurons under different layers is shown in Figure 9(a), which is helpful to further obtain the optimal layers of Simple RNN model.

For a clearer display, only the accuracy results of training epochs from 350 to 500 are shown in Figure 9(a). As can be seen from the figure, although the accuracy of 2-layer and 1-layer is relatively stable, the average accuracy is still lower than 4-layer model. Therefore, 4-layer model is considered to be the optimal layers of Simple RNN model with an average accuracy of 97.5%. Figure 9(b) illustrates the accuracy of 4-layer Simple RNN with 16 neurons under different LRs, from which we can see that the accuracy curve for LR of 0.005 is always in an unconvergent state. In addition, it is easy to find that when LR is 0.001, the accuracy curve of the model gradually converges with the increasing of training epoch, and finally achieve the highest accuracy, which is about 98%. Therefore, considering the above experimental results, when LR of 4-layer Simple RNN with 16 neurons is 0.001, the best performance of the model can be achieved.

B. DETECTION RESULTS OF ABNORMAL BEHAVIOR FOR UAV

It is worth mentioning that the normal behavior models applied in this paper are the ready models trained based on DOA estimated angle data. Therefore, the NRMSE between the current moment position of UAV and the position information provided by normal behavior models can be obtained in real time, thus achieving real-time detection. Specifically, the specific calculation process of NRMSE is as follows:

- Step 1: Complete the normalized calculation of the angle data,

$$X_s = \frac{(X - X.min) \cdot (max - min)}{(X.max - X.min)} + min \tag{16}$$

  where X represents all samples of the angle data, $X.min$ and $X.max$ represent the minimum and maximum of each column. Since the normalized maximum value is 1 and minimum value is 0. Therefore, $max = 1$ and $min = 0$.

- Step 2: Complete the calculation of RMSE,

$$RMSE(X_s) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}[X_s(i) - \hat{X}_s(i)]} \tag{17}$$

  where m is the total number angle data columns, which is equal to 2 in this paper (elevation angle θ and horizontal angle ϕ), $\hat{X}_s(i)$ is the data obtained from trained normal behavioral models.

Besides, in order to ensure the accuracy of the abnormal behavior detection scheme for UAV, this paper adopts the singular value detection method. Specifically, when the threshold is too large, it means that there is a greater tolerance for the deviation of UAV from the normal behavior model, which will cause some abnormal behaviors of UAV to be undetected normally, thus reducing the accuracy of the detection scheme; When the threshold is too small, it means that the tolerance range of UAV's deviation from the normal behavior model is too small, and the detection scheme will incorrectly detect the situation that UAV is offset by the external air pressure to be abnormal, which will affect the performance of the detection scheme. Therefore, this paper needs a lot of experiments to choose the most appropriate threshold,

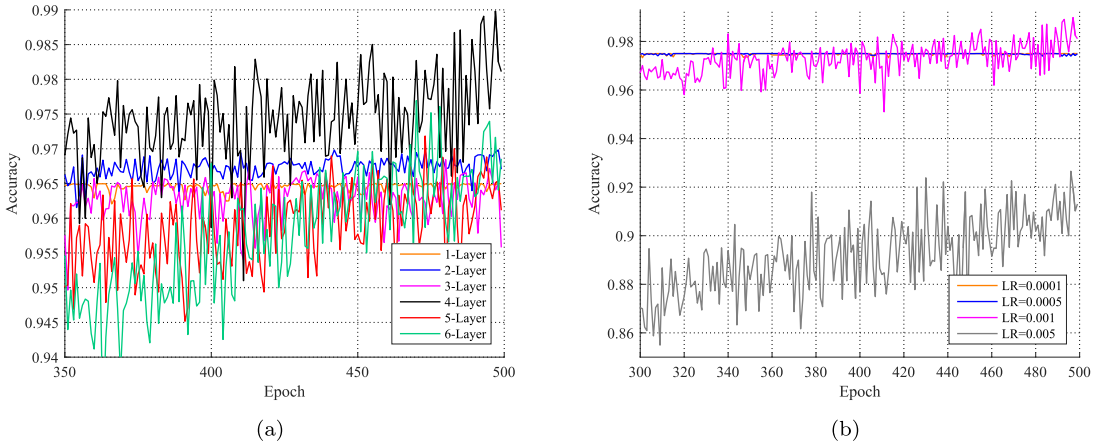


FIGURE 9. Accuracy of Simple RNN under different layers and LRs of highway scenario. (a) Accuracy under different layers. (b) Accuracy under different LRs.

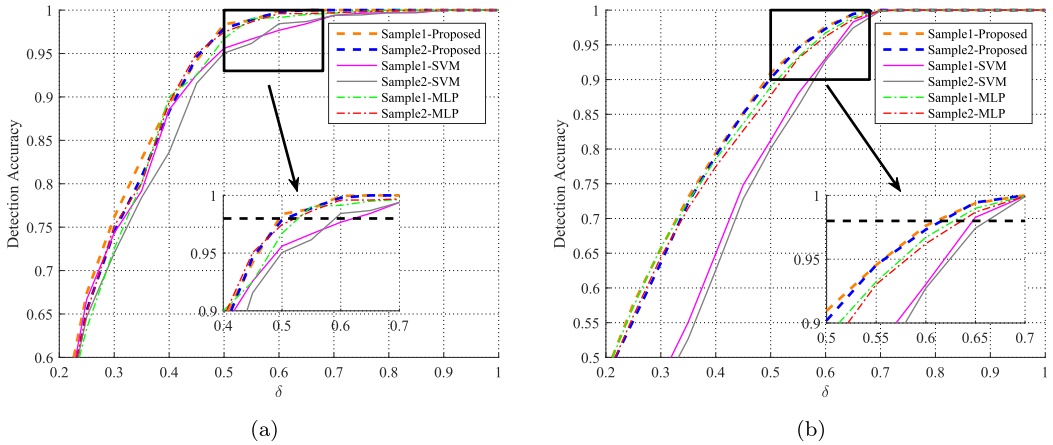


FIGURE 10. Detection accuracy and comparison with SVM and MLP under two scenarios.(a) Detection accuracy and comparison in smart city scenario. (b) Detection accuracy and comparisonin highway scenario.

and multiple samples and different thresholds are used for testing. In addition, in order to make the display of the image clearer, two samples of proposed models (LSTM-RNN model in smart city and Simple RNN model in highway) are randomly selected to conduct the experiments, and also compare the performance with Support Vector Machine (SVM) and Multi-Layer Perception (MLP) to prove the reliability of the proposed method [28], the experimental results are shown in Figure 10 (a) and (b), respectively.

Figure 10(a) shows the detection accuracy under different thresholds in smart city scenario. It can be seen from the figure that as the threshold increases, all accuracy curves show a gradual upward trend. In order to ensure the accuracy of abnormal behavior detection, δ is defined as 0.5, which can tolerate the slight deviation of UAV when it is affected by the air pressure, and accurately detect the abnormal behavior. Besides, when the δ is set to 0.5, the detection accuracy is about 98%. For SVM and MLP algorithm, two samples are also randomly selected for comparison. It can be seen from the figure that when the accuracy of SVM curves reaches 98%, the corresponding threshold is about 0.62, and when the accuracy of MLP curves reaches 98%, the corresponding threshold is about 0.58, both of which are larger than 0.5 of proposed scheme. Therefore, under the condition of ensuring the same accuracy, the abnormal behavior detection scheme proposed in this paper is more sensitive with a lower false positive rate of detection in smart city scenario. Figure 10(b) illustrates the detection accuracy under different thresholds in highway scenario, and the δ is defined as 0.6 to achieve the high detection accuracy. Besides, the detection accuracy of 0.6 is also about 98%, when the accuracy of SVM curves reaches 98%, the corresponding threshold is about 0.64, and when the accuracy of MLP curves reaches 98%, the corresponding threshold is about 0.63, both of which are also larger than 0.6 of proposed scheme. Therefore, the abnormal behavior detection scheme in highway scenario is also more sensitive with a lower false positive rate of detection compared with SVM method.

In addition, it is necessary to provide the computation time of proposed method, SVM and MLP algorithms in test stage and run stage to further prove the feasibility of proposed abnormal behavior detection scheme. Specifically, the computation time in test stage refers to the time taken to test the model using test sets after the model is built, which also includes the loading of model, reading of the data and the abnormal judgment. And the computation time in test stage of proposed method, SVM and MLP algorithms is 129.85s, 130.59s and 132.97s, respectively. Besides, as the most important basis, the computation time in run stage (which is the time required to complete the abnormal detection) of proposed method, SVM and MLP is 0.034s, 0.035s and 0.033s, respectively. Therefore, the abnormal behavior detection scheme proposed in this paper can not only achieve more accurate detection results, but also need to spend almost the same or even shorter detection time.

V. SUMMARY AND FUTURE WORK

This paper proposes an abnormal behavior detection scheme to address the problem of GPS spoofing. From the works of predecessors, it is clear that deep learning performs well in prediction by applying trained model. Inspired by this point, an abnormal behavior detection scheme of UAV using RNNs is proposed in this paper. To avoid the confusion of slight offset and abnormal behavior of UAV, the reliable normal behavior models for two different scenarios including smart city and highway are established by applying RNNs, and DOA estimation algorithm is used to collect current angle data of UAV thereby ensuring the accuracy of input datasets. According to the experimental results, the optimal normal behavior models under smart city and highway scenarios are LSTM-RNN and Simple RNN with highest accuracy. Moreover, an appropriate threshold is selected through amounts of experiments to measure NRMSE between the real position and the position provided by normal behavior models, so as to achieve the purpose of detecting abnormal behavior. Experimental results justify that the proposed abnormal behavior scheme is of high detection accuracy, then the abnormal behavior of UAV can be detected in time, which provides the possibility for people to take remedial measures, thus ensuring the security of the communication system. In the future, the work can be extended to optimize the simulation system of experiments and make it more in line with the real unstable environment of UAV, and the proposed normal behavior model can also be upgraded to deal with very large scale angle datasets.

REFERENCES

[1] H. Wang, J. Wang, J. Chen, Y. Gong, and G. Ding, "Network-connected UAV communications: Potentials and challenges," *China Commun.*, vol. 15, no. 12, pp. 111–121, Dec. 2018.

[2] S. ur Rahman, G. Kim, Y. Cho, and A. Khan, "Positioning of UAVs for throughput maximization in software-defined disaster area UAV communication networks," *J. Commun. Netw.*, vol. 20, no. 5, pp. 452–463, Oct. 2018.

[3] J. Kakar, A. Chaaban, V. Marojevic, and A. Sezgin, "UAV-aided multi-way communications," in *Proc. IEEE 29th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2018, pp. 1169–1173.

[4] Q. Zou, S. Huang, F. Lin, and M. Cong, "Detection of gps spoofing based on uav model estimation," in *Proc. 42nd Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2016, pp. 6097–6102.

[5] Y. Qiao, Y. Zhang, and X. Du, "A vision-based GPS-spoofing detection method for small UAVs," in *Proc. 13th Int. Conf. Comput. Intell. Secur. (CIS)*, Dec. 2017, pp. 312–316.

[6] S. Weiss, D. Scaramuzza, and R. Siegwart, "Monocular-SLAM–based navigation for autonomous micro helicopters in GPS-denied environments," *J. Field Robot.*, vol. 28, no. 6, pp. 854–874, 2011.

[7] G. Chowdhary, E. N. Johnson, D. Magree, A. Wu, and A. Shein, "GPS-denied indoor and outdoor monocular vision aided navigation and control of unmanned aircraft," *J. Field Robot.*, vol. 30, no. 3, pp. 415–438, 2013.

[8] K. D. Wesson, B. L. Evans, and T. E. Humphreys, "A combined symmetric difference and power monitoring GNSS anti-spoofing technique," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Dec. 2013, pp. 217–220.

[9] L. He, H. Li, W. Li, and M. Lu, "Neural network based C/N0 abnormity detection method for GPS anti-spoofing," in *Proc. Int. Tech. Meeting Inst. Navigat.*, 2016, pp. 716–725.

[10] E. Falletti, B. Motella, and M. T. Gamba, "Post-correlation signal analysis to detect spoofing attacks in GNSS receivers," in *Proc. 24th Eur. Signal Process. Conf. (EUSIPCO)*, 2016, pp. 1048–1052.

[11] H. Li and X. Wang, "Detection of GPS spoofing through signal multipath signature analysis," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2016, pp. 1–5.

[12] M. Das and S. K. Ghosh, "A deep-learning-based forecasting ensemble to predict missing data for remote sensing analysis," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 12, pp. 5228–5236, Dec. 2017.

[13] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems," *IEEE Access*, vol. 6, pp. 37328–37348, 2018.

[14] I. Choi, H. Song, and J. Yoo, "Deep learning based pedestrian trajectory prediction considering location relationship between pedestrians," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, 2019, pp. 449–451.

[15] B. Li, X. Li, and W. Wang, "Research on barometric altimeter aiding GPS arithmetic in challenge environment," in *Proc. 13th IEEE Int. Conf. Electron. Meas. Instrum.*, Oct. 2017, pp. 581–585.

[16] X. Lan, Y. Li, and Y. Chen, "A novel 2D DOA estimation via tensor modeling for cylindrical conformal array," in *Proc. 9th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, 2017, pp. 1–6.

[17] X. Lan, L. Wang, Y. Wang, C. Choi, and D. Choi, "Tensor 2-D DOA estimation for a cylindrical conformal antenna array in a massive MIMO system under unknown mutual coupling," *IEEE Access*, vol. 6, pp. 7864–7871, 2018.

[18] Y. B. Nechaev, I. W. Peshkov, and N. A. Fortunova, "Cylindrical antenna array development and measurements for DOA-estimation applications," in *Proc. 11th Int. Conf. Antenna Theory Techn. (ICATT)*, 2017, pp. 155–158.

[19] P. Yu and L. Wang, "Wideband DOA estimation by using off-grid technique for a cylindrical conformal array," in *Proc. CIE Int. Conf. Radar*, 2017, pp. 1–4.

[20] P. Yang, F. Yang, and Z. Zai-Ping, "DOA estimation of cylindrical conformal array by MUSIC algorithm," *J. Radio Sci.*, vol. 23, no. 2, pp. 288–291, 2008.

[21] H. Zhao, M. Cai, and H. Liu, "Two-dimensional DOA estimation with reduced-dimension MUSIC algorithm," in *Proc. Int. Appl. Comput. Electromagn. Soc. Symp. (ACES)*, Aug. 2017, pp. 1–2.

[22] R. Li, L. Xu, X.-W. Shi, L. Chen, and C.-Y. Cui, "Two-dimensional NC-music DOA estimation algorithm with a conformal cylindrical antenna array," *J. Electromagn. Waves Appl.*, vol. 25, nos. 5–6, pp. 805–818, 2011.

[23] D. Aksu and M. A. Ayadin, "Detecting port scan attempts with comparative analysis of deep learning and support vector machine algorithms," in *Proc. Int. Congr. Big Data, Deep Learn. Fighting Cyber Terrorism (IBIGDELFT)*, 2018, pp. 77–80.

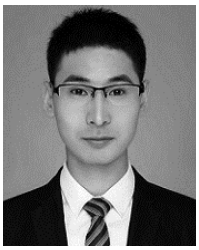
[24] X. Wei, Y. Liu, S. Gao, X. Wang, and H. Yue, "An RNN-based delay-guaranteed monitoring framework in underwater wireless sensor networks," *IEEE Access*, vol. 7, pp. 25959–25971, 2019.

[25] T. Ergen and S. S. Kozat, "Online training of LSTM networks in distributed systems for variable length data sequences," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 5159–5165, Oct. 2018.

[26] Y. Yuan, C. Tian, and X. Lu, "Auxiliary loss multimodal GRU model in audio-visual speech recognition," *IEEE Access*, vol. 6, pp. 5573–5583, 2018.
[27] M. Reza, S. Kumar, and L. Hussain, "Energy load forecasting using deep learning approach-LSTM and GRU in spark cluster," in *Proc. 5th Int. Conf. Emerg. Appl. Inf. Technol.*, 2018, pp. 1–4.
[28] Y. Yang, J. Wang, and Y. Yang, "Exploiting rotation invariance with SVM classifier for microcalcification detection," in *Proc. 9th IEEE Int. Symp. Biomed. Imag. (ISBI)*, May 2012, pp. 590–593.

KE XIAO received the Ph.D. degree from the School of Information Science and Technology, North China University of Technology. He is currently a Visiting Scholar with the University of Washington. His main research interests include wireless communications, physical layer security, the Internet of Things, and embedded systems.

JIANYU ZHAO is currently pursuing the master's degree with the School of Information Science and Technology, North China University of Technology. Her main research interests include artificial intelligence (AI), wireless communication, and the Internet of Things.



YUNHUA HE received the Ph.D. degree from the School of Information Science and Technology, North China University of Technology. His main research interests include security and privacy in cyber-physical systems, Bitcoin-based incentive mechanism, and security and privacy in vehicle ad hoc networks.

CHAOFEI LI is currently pursuing the master's degree with the School of Information Science and Technology, North China University of Technology. His main research interests include artificial intelligence (AI), wireless communication, and the Internet of Things.

WEI CHENG received the Ph.D. degree from the School of Engineering and Technology, University of Washington. His main research interests include security and cyber-physical systems, HCI-based security, location privacy protection, smart cities, and underwater networks.