DroneMA: Drone Mobility Alignment Countering AI-based Spoofing Attacks

Weiyang Li[1], Ning Wang[1*], Chuan Ma[1], Tao Xiang[1], Kai Zeng[2]
[1]College of computer science, Chongqing University, China
[2]Department of Electrical and Computer Engineering, George Mason University, USA
weiyangli@stu.cqu.edu.cn, {nwang5, chuan.ma, txiang}@cqu.edu.cn, kzeng2@gmu.edu

***Abstract*—The rapid advancement of generative AI models empowers adversaries to craft sophisticated spoofing signals that closely mimic legitimate transmissions at the physical layer. This development poses a significant challenge for UAV spoofing detection, particularly due to the severe constraints on drone resources. To address this challenge, we propose a novel mobility alignment-based spoofing detection framework for drones, termed DroneMA. This method leverages an observed phenomenon: the relative positions and movements of a drone with respect to an attacker and a legitimate ground control station (GCS) are inherently different. Consequently, the mobility characteristics of spoofing signals differ from those of legitimate signals. In DroneMA, the variation trend in the received signal strength indicator (RSSI) is employed to capture the mobility characteristics of drones, thereby facilitating practical implementation on off-the-shelf devices. To mitigate the inaccuracies inherent in RSSI, we introduce a feature alignment method that utilizes Z-score normalization and a Gated Recurrent Unit (GRU) network. Addressing the challenge of unpredictable attack models, we propose a detection scheme based on the Interquartile Range (IQR) method, enabling the development of an effective identification model using only positive samples. We conducted real-world experiments to evaluate the effectiveness of DroneMA, demonstrating a remarkable average accuracy of 92.78% across three typical flight scenarios.**

I. Introduction

Recent years have witnessed a spurt of progress in unmanned aerial vehicles (UAVs), and consumer UAVs, commonly known as drones, have demonstrated significant potential and value in various domains, including smart agriculture, public security, intelligent logistics, and military operations [1], [2]. In these application scenarios, drone security authentication, particularly the verification, and authentication of received control commands on the drone's end, is of paramount importance [3], [4]. However, due to the severe constraints on drones' energy and computational resources, it is challenging to implement complex algorithmic models and extensive authentication interactions. This limitation results in a significant vulnerability in the verification of received signals. Exploiting this weakness, malicious attackers can use pre-acquired information, such as the source and destination IP/MAC addresses, preamble, and protocol structure, to generate spoofing signals, thereby disrupting and commandeering the drones [5]. Therefore, spoofing attacks pose a serious threat to drone communications.

* Corresponding author: Ning Wang (Email: nwang5@cqu.edu.cn).

Unfortunately, the rise of generative artificial intelligence (AI) technology in recent years has further exacerbated concerns regarding drone security detection and authentication. Leveraging powerful generative AI models, malicious attackers can easily mimic legitimate signals and even generate spoofing signals with similar physical layer characteristics [6], [7]. By relying on a period of sniffing, AI-based attackers are allowed to generate high-quality spoofing signals without prior knowledge of specific drone information. These spoofing signals exhibit highly deceptive physical layer characteristics, rendering traditional physical layer and higher-layer security authentication methods ineffective. This emerging AI-based spoofing attack poses a severe security challenge to current drone security detection systems, especially given the significant resource constraints of drones.

In general, physical layer authentication emerges as a promising technology to counter spoofing attacks [8]–[10]. However, traditional physical layer security authentication techniques prove inadequate when addressing these new AI-based spoofing attacks in drone application scenarios. Existing physical layer security authentication methods can be categorized into two types: physical layer authentication methods based on channel state [11]–[13] and hardware identification [14]–[16] using RF fingerprints. channel state-based physical layer authentication proves unsuitable for high-mobility drone environments [17], as drones' dynamic nature significantly undermines channel reciprocity's manifestation. RF fingerprint-based physical layer authentication, which depends on subtle differences in physical layer characteristics of RF signals, often requires additional signal extraction and analysis techniques, posing challenges for scalability to off-the-shelf devices [18]. Furthermore, generative AI models equip spoofing attackers with the ability to produce spoofing signals embedded with counterfeit physical layer features, thereby compromising the drone's capacity to discern the physical characteristics of the RF fingerprinting [6], [7]. As a result, considering the high mobility and resource constraints of drones, developing lightweight detection methods capable of countering generative AI model-based spoofing attacks remains an open problem.

To fill this gap, this paper proposes a novel mobility alignment-based spoofing detection scheme for drones, called DroneMA, that leverages the mobility alignment reflected in both the communication and sensing systems of drones.

By focusing on the dynamic nature of drone movement and the corresponding RSSI variations, our framework transforms the spoofing detection task into a problem of detecting inconsistencies in mobility patterns. This approach provides a robust and real-time solution tailored specifically for drone communication scenarios by overcoming the challenges of using RSSI as a detection feature, including its sensitivity to environmental disturbances in drones, the impact of Automatic Gain Control (AGC), and Global Positioning System (GPS) localization inaccuracies, thereby ensuring significant performance with easily extractable features. Extensive experiments in real-world environments were conducted to validate the feasibility and effectiveness of our proposed scheme. The experimental results demonstrate that our scheme can achieve accuracy rates of 95.85%, 87.62%, and 96.43% in three typical drone operation scenarios, respectively. The main contributions of this paper are summarized as follows:

- We propose a novel physical layer security authentication method that exploits the consistency in a drone's mobility attributes across different modalities. By leveraging the inherent differences in the relative positions and movements of the drone with respect to an attacker and a legitimate GCS, we develop a spoofing detection approach based on drone mobility.
- We present a novel spoofing attack detection framework based on mobility alignment, termed DroneMA, which utilizes easily extractable RSSI to characterize drone mobility, facilitating its extension to off-the-shelf devices. It also addresses RSSI instability issues through Z-score normalization and an RSSI-to-Distance prediction network model based on GRU (R2D-GRU) network. Additionally, an IQR-based detection method is employed to train the detection model using only positive samples.
- We conduct real-world proof-of-concept experiments using off-the-shelf devices under different scenarios and test the proposed detection scheme.

Furthermore, the proposed DroneMA scheme has the following salient features:

- **Real-Time Detection**: Operates in real-time, continuously monitoring communication and sensing data from the drone to promptly detect any spoofing attacks.
- **Lightweight**: Designed to be computationally efficient, making it suitable for deployment on resource-constrained drone platforms without overwhelming the drone's processing capabilities.
- **Adaptability**: Designed for typical drone-ground communication scenarios, this approach can be readily extended to related applications.
- **Low Deployment Cost**: Utilizes easily accessible RSSI and positional data from standard drone communication and sensing systems, eliminating the need for additional hardware and simplifying deployment.

In the remainder of this article, we first introduce related work in Section II. The system model, motivation, and challenges are given in Section III. Section IV presents the proposed schemes. Section V provides the experiments and results, and finally, Section VI concludes this paper.

II. Related Work

Identity spoofing attacks pose significant threats to wireless communication systems. In recent years, the rapid development of generative AI technology has endowed spoofing attacks with the ability to forge physical layer features, making them more covert and harmful. Traditional security approaches rely on cryptographic authentication. However, physical layer spoofing attackers may have already captured critical identification information from legitimate communications through preliminary sniffing, thereby compromising the security of cryptographic authentication mechanisms. Consequently, physical-layer security has emerged as a promising technique to complement and enhance cryptographic mechanisms. Current physical-layer spoofing attack detection schemes can be mainly divided into channel state-based and hardware identification-based.

channel state-based methods utilize the inherent spatial characteristics of wireless channels to detect spoofing attacks, such as channel state information (CSI) [11] and received signal strength (RSS) [12], [13]. However, in mobile scenarios, the measurement results vary over time, leading to excessive false positives. Consequently, these methods do not apply to drone-GCS systems. Hardware identification-based methods leverage the inherent imperfections that arise during the manufacturing process of hardware components to uniquely verify the identity of devices, such as Carrier Frequency Offset (CFO) [14], I/Q imbalance [15], and clock skew [16]. However, the physical layer features derived from hardware impairments are often subtle and hard to extract, rendering them ineffective for detecting spoofing attacks.

Recently, researchers have begun to utilize physical-layer methods to counter identity spoofing attacks in drone systems. Some works design cryptographic schemes utilizing the physical-layer characteristics of drones [19], [20], enhancing the security of drone communications. However, these schemes can negatively impact the real-time responsiveness of resource-constrained drones. Other works have designed novel physical-layer authentication (PLA) frameworks [21]–[23], which mainly rely on simulations rather than real drone communication data, raising concerns about their effectiveness and reliability in real-world scenarios.

Most notably, none of the aforementioned studies have addressed AI-based spoofing attacks, which are capable of generating spoofing signals that closely replicate the physical layer characteristics of legitimate signals, thereby significantly enhancing the likelihood of bypassing existing physical-layer authentication methods. In contrast to these existing approaches, our scheme focuses on detecting spoofing attacks by leveraging the mobility characteristics reflected in both the communication and sensing systems of drones. While it employs signal strength as a feature, it diverges from traditional channel state-based methods by emphasizing mobility alignment within the drone model. Our solution, which

performs detection from the drone's perspective, is specifically designed for drone communication scenarios, offering easily extractable features, notable performance improvements, and real-time continuous spoofing attack detection. Crucially, it addresses AI-based spoofing attacks by capitalizing on the relative mobility of the drone—an aspect that has been previously overlooked in the literature.

III. System Model, Motivation and Challenge

A. System model

1) Drone system model: We consider a typical drone model that includes a control system, a propulsion system, a communication system, and a sensing system.

The control system primarily consists of the Flight Controller, which is responsible for attitude control, mission planning, and data processing. The propulsion system encompasses the Motors, which are governed by the Flight Controller to facilitate the drone's movement. The communication system includes the Remote Control (RC) Radio and the Telemetry Radio. The RC Radio allows interaction with the remote controller, while the Telemetry Radio connects to the GCS using a specific communication module and protocol, providing status updates on this link. The sensing system consists of Sensors, with Inertial Measurement Unit (IMU) and GPS receivers being the most common. These sensors monitor crucial state information about the drone and its environment, providing data on the drone's speed, attitude, and position.

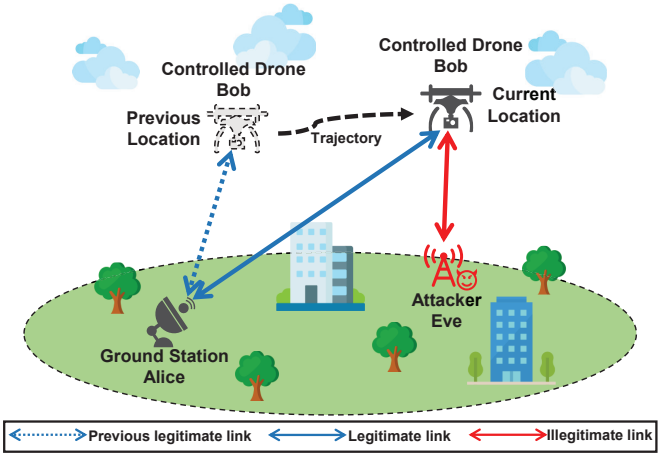


Fig. 1: System model.

2) Drone-GCS communication system model: Referring to previous studies [23], [24], we consider a typical three-entity model to describe the drone-GCS communication link, as shown in Fig. 1. This model includes a legitimate GCS Alice, a drone Bob, and a malicious attacker Eve. Specifically, the drone is controlled by the ground station, while the malicious attacker executes control signal spoofing attacks by sending deceptive control signals in the same format. Due to the open nature of the wireless transmission medium, drones, and drone-GCS communication systems are susceptible to spoofing attacks. Once a malicious attacker masquerades as the ground station to access the drone-GCS network, they can launch man-in-the-middle and physical collision attacks, causing severe damage. Drones typically operate at relatively high altitudes, while GCS is generally located on the ground. Due to the high elevation of drones and the typically unobstructed environment, the communication between the drone and the GCS is predominantly through line-of-sight (LOS) channels.

3) Attack model: We assume a powerful AI-based attacker can masquerade the legitimate GCS by modifying its own identity into the legitimate user's. The attacker can manipulate arbitrary fields in a frame, such as the source and destination IP/MAC addresses, sequence numbers, frame checks, and so on. It can leverage AI generative models to generate spoofing signals with fabricated physical layer characteristics after sniffing the communication between the drone and GCS for enough time. During the execution of the attack, the attacker maintains a stationary position, and its location can be arbitrary, but typically it avoids being too close to the GCS to prevent arousing suspicion. Besides, the attacker typically analyzes the drone's signals after detecting its presence before launching an attack. Consequently, attacks are generally not initiated during the initial moments of communication between the drone and the GCS. It is worth noting that GPS spoofing attack detection for drones is beyond the scope of this discussion. To simplify the analysis, we assume that GPS signals possess spoofing attack detection capabilities, meaning that the GPS information is considered to be authentic and reliable.

4) Security obective: Our approach aims to detect spoofing attacks by continuously monitoring the data from the drone's communication system and sensing system in real-time. This enables the drone to take emergency measures such as returning to the home position or switching communication methods if an attacker impersonates the GCS. Since communication with GCS becomes ineffective during a spoofing attack, this detection method needs to be deployed on the drone itself. Specifically, we set a sliding time window of a certain length to capture the drone's state data from the current moment and the recent past. Our approach is conducted after establishing the communication link between the drone and GCS. As spoofing attacks generally do not occur during the initial phase of communication, we can establish a continuous detection sliding window during the communication initialization phase to enable real-time continuous detection of spoofing attacks.

B. Motivation

Our spoofing attack detection schemes are motivated by the following observations.

1) The communication and sensing systems both can reflect drone movement trends: To maintain stable flighting, drones rely on sensors such as GPS and IMU for real-time state estimation. These sensors enable drones to continuously obtain their positional coordinates and record the GCS's location at takeoff, thereby reflecting their movement trends relative to the GCS in real-time. Additionally, the drone's communication system, which primarily receives task commands and transmits remote sensing information, also records the current RSSI to reflect the drone-GCS communication quality in real-time.

RSSI is a measure of the power level that a wireless device receives from a signal. The RSSI received by terminals is often affected by signal reflection, diffraction, and shadowing, causing discrepancies between the actual radio loss and theoretical predictions. Researchers commonly use the log-distance path loss model (LDPL) to describe the relationship between RSSI in dB and distance. According to the LDPL, there is a logarithmic relationship between RSSI and distance, meaning that as the distance between the transmitter and receiver increases, the RSSI decreases logarithmically. The LDPL model can be expressed by the equation above:

$$\text{RSSI} = \text{RSSI}_0 - 10 \cdot n \cdot \log_{10}\left(\frac{d}{d_0}\right), \tag{1}$$

where RSSI_0 is the RSSI value at the reference distance d_0, n is the path loss exponent that depends on the environment (typically 2 for free space, higher for urban or indoor environments), d is the distance between the transmitter and receiver, d_0 is a reference distance, usually 1 meter. The path loss model is commonly used for distance estimation in indoor localization tasks, combined with filtering techniques. However, for outdoor scenarios, especially those involving highly mobile entities, directly using the path loss model to predict distance can result in unacceptable errors. During flight missions, drones exhibit high mobility, and the RSSI received by the drone changes continuously as it moves. Although we cannot directly compute the distance, this phenomenon allows us to infer the trend of the drone's distance relative to the GCS over time using the RSSI series. This movement trend should be highly consistent with that reflected by the sensing system.

2) Easy access to RSSI: Unlike other hardware physical-layer characteristics (such as CFO, I/Q imbalance, and CSI), RSSI is the most readily accessible one. It can be easily obtained from existing communication hardware without requiring additional hardware support or computational overhead. In typical drone-GCS communication links, RSSI values are transmitted to the flight controller via the communication protocol, allowing easy recording of both receiver and transmitter RSSI at any time. Consequently, RSSI-based spoofing attack detection schemes for drones are highly versatile and flexible.

3) Differences in Relative Movement Between the Drone, Legitimate GCS, and Attacker: Due to the covert nature of wireless signal spoofing attacks, the legitimate GCS and the attacker are often not co-located. This geographical disparity creates differences in the relative movement between the drone and the legitimate GCS compared to the drone and the attacker, which can be leveraged for detecting spoofing attacks. Although AI-based spoofing attacks generate signals that closely resemble legitimate ones, leading to a certain degree of ineffectiveness in traditional physical-layer authentication methods, they cannot overcome the challenges posed by geographical and movement discrepancies. The GCS's location is difficult to acquire, and the drone's movement is unpredictable. Even if an attacker knows the position of the GCS and the drone's future movements, replicating the corresponding RSSI is extremely challenging due to its inherent randomness.

Based on these observations, we propose leveraging mobility alignment to detect spoofing attacks on drones. Specifically, while the drone's sensing system computes the distance between the drone and the GCS, the RSSI also partially reflects this distance. Therefore, if the communication originates solely from the legitimate GCS, the distance indicated by the RSSI should exhibit a consistent trend with the distance calculated by the sensing system. In contrast, if a spoofing attack is present, this relationship will be disrupted.

C. Challenges

Although RSSI is related to distance and can reflect drone movement, achieving effective spoofing attack detection for drone-GCS networks is non-trivial. The following challenges must be addressed:

1) Environmental and Temporal Variability: RSSI values are influenced by environmental factors such as temperature, humidity, and random noise, causing fluctuations even at a constant distance over time. This complicates establishing a consistent correlation between RSSI and distance.

2) Impact of Automatic Gain Control (AGC): In drone radio systems, RSSI values are primarily used to assess the quality of wireless communication. The AGC circuit dynamically adjusts the gain based on these RSSI values, which means that the RSSI values we obtain are influenced by the AGC algorithm. Consequently, this can lead to instances where longer distances result in higher RSSI values compared to shorter distances for the same pair of radios.

3) Inaccuracy of GPS Positioning: Drones rely on navigation systems for positional information, with GPS-IMU being the most prevalent. GPS accuracy is limited by meter-level errors, and altitude estimation can suffer from even greater errors. This imprecision affects the drones' ability to obtain accurate distance measurements, impacting the effectiveness of RSSI-based spoofing attack detection.

IV. DroneMA Framework

To address the main issues discussed above, in this section, we propose a novel detection framework, i.e., Drone Mobility Alignment Countering AI-based Spoofing Attacks.

A. Overview

Our framework is designed to operate on drones during flight missions, as spoofing attacks often occur during this time. By leveraging the mobility characteristics reflected in both the drone communication system and sensing system, our framework transforms the task of drone-GCS communication spoofing detection into the detection of mobility alignment.

Fig. 2 shows an overview of the proposed scheme, where the DroneMA can be divided into three parts: Z-score Normalization, R2D-GRU network, and IQR-based detection. While in flight, the drone's communication system and sensing system continuously record the RSSI data and the distance to the GCS at each moment. The collected real-time flight data is then processed using these components to detect spoofing attacks.

A fixed-length sliding window of size T, sliding with a step size of 1, records all RSSI and distance data over the past T

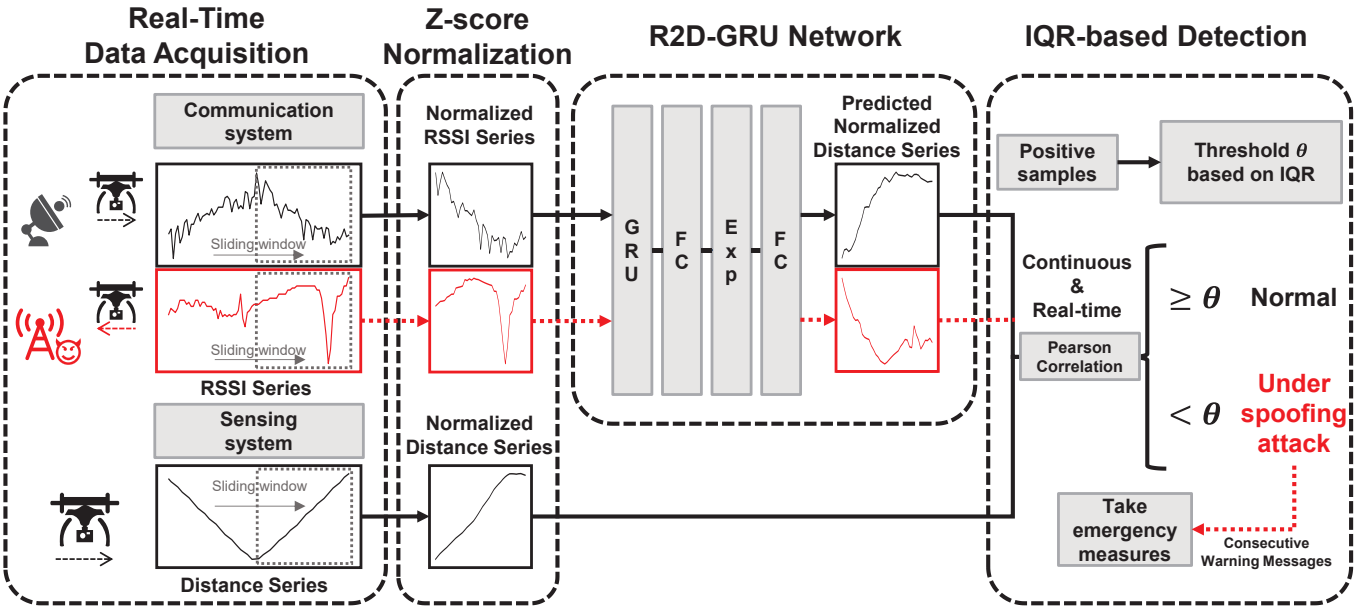


Fig. 2: DroneMA framework. Z-score normalization preprocesses the real-time RSSI and distance series collected by the drone. R2D-GRU network then predicts the corresponding distance series from the normalized RSSI data. IQR-based detection method compares the predicted and actual distance values using a threshold θ to identify spoofing attacks. If consecutive warnings are detected, the drone takes emergency measures to prevent further harm.

moments. For simplicity, all RSSI values are denoted as R and all distance values as D. At time t_c, the RSSI series and distance series are:

$$\mathbf{R}_{t_c} = [R_{t_c-T+1}, R_{t_c-T+2}, \ldots, R_{t_c}], \tag{2}$$

$$\mathbf{D}_{t_c} = [D_{t_c-T+1}, D_{t_c-T+2}, \ldots, D_{t_c}]. \tag{3}$$

In the DroneMA Framework, the RSSI series and distance series first undergo Z-score normalization as a preprocessing step. The normalized RSSI series is then fed into the R2D-GRU network to predict the normalized distance series. Finally, the predicted normalized distance series is compared with the actual normalized distance series to compute the correlation. An IQR-based threshold is used to determine the presence of a spoofing attack. Next, we will provide a detailed explanation of each component of the framework.

B. Z-score normalization

Given the difficulties in directly predicting distance values using RSSI due to various factors mentioned in the challenges, we adopt a strategy of predicting distance variation trends using RSSI variation trends. Specifically, we apply Z-score normalization to the RSSI values and distance values. The Z-score normalization is calculated using the formula:

$$R'_t = \frac{(R_t - \mu_{R_T})}{\sigma_{R_T}}, \tag{4}$$

$$D'_t = \frac{(D_t - \mu_{D_T})}{\sigma_{D_T}}, \tag{5}$$

where R_t and D_t are the respective values at time t, R'_t and D'_t represent the normalized RSSI and distance values at time t, μ_{R_T} and μ_{D_T} are the mean values, given by $\mu_{R_T} = \frac{1}{T}\sum_{i=1}^{T} R_i$ and $\mu_{D_T} = \frac{1}{T}\sum_{i=1}^{T} D_i$, and σ_{R_T} and σ_{D_T} are the standard deviations, given by $\sigma_{R_T} = \sqrt{\frac{1}{T}\sum_{i=1}^{T}(R_i - \mu_{R_T})^2}$ and $\sigma_{D_T} = \sqrt{\frac{1}{T}\sum_{i=1}^{T}(D_i - \mu_{D_T})^2}$, of the RSSI and distance values over the time period T. At this point, we can obtain the normalized RSSI series and distance series $\mathbf{R}'_{t_c}$ and $\mathbf{D}'_{t_c}$.

C. R2D-GRU network

To address the limitations of traditional RSSI-based distance estimation methods in drone scenarios, such as their inability to adapt to dynamic outdoor environments and lack of precision, we designed an RSSI-to-Distance prediction network model based on GRU (R2D-GRU Network). Furthermore, since the network focuses on Z-score normalized RSSI and distance, it can exhibit a certain level of generalizability and does not require retraining for specific communication modules. As illustrated in Fig. 3, the network takes as input the normalized RSSI series ($\mathbf{R}'_{t_c}$). Each time step's normalized RSSI value (R'_t) is fed into the network to predict the corresponding normalized distance ($\hat{D}'_t$) for that specific time step. The network then outputs a normalized distance series ($\hat{\mathbf{D}}'_{t_c}$) of the same length as the input, which is used for subsequent spoofing attack detection. In this way, the R2D-network achieves the prediction of distance variation trends based on RSSI variation trends. The R2D-GRU network consists of five layers: an input layer, a GRU layer, a hidden layer, an exponential layer, and an output layer.

1) Input layer: The input layer receives the normalized RSSI series ($\mathbf{R}'_{t_c}$) and sequentially passes each time step's normalized RSSI (R'_t) to the next layer.

2) GRU Layer: The Gated Recurrent Unit (GRU) layer comprises multiple GRU units that process time series through

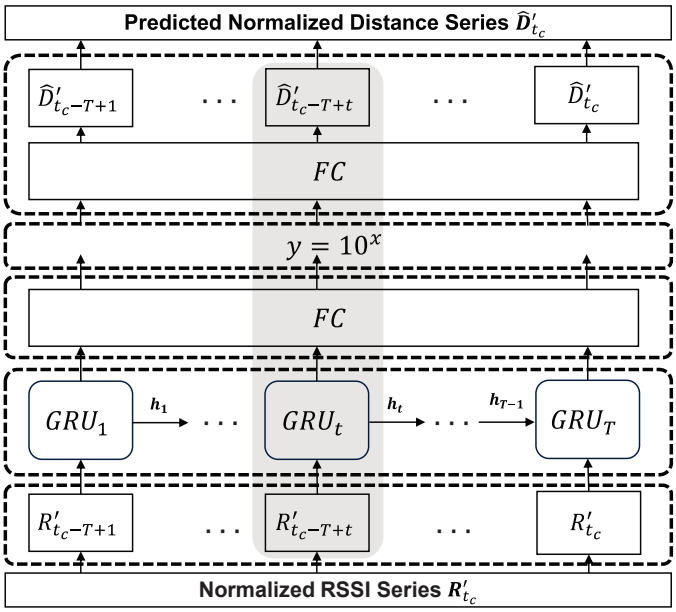


Fig. 3: Time-Unfolded architecture of R2D-GRU network. The gray box indicates the network's processing of data at a specific time point.

update and reset gates, capturing long-term dependencies. The GRU uses two main gates: the update gate and the reset gate. The update gate z_t determines how much of the previous state h_{t-1} is retained, while the reset gate $reset_t$ controls how much of the past information is forgotten:

$$z_t = \sigma(W_{zx}x_t + W_{zh}h_{t-1} + b_z), \tag{6}$$

$$reset_t = \sigma(W_{rx}x_t + W_{rh}h_{t-1} + b_r), \tag{7}$$

$$\tilde{h}_t = \tanh(W_{hx}x_t + W_{hh}(reset_t \odot h_{t-1}) + b_h), \tag{8}$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t, \tag{9}$$

where x_t is the input vector, h_t is the output vector, z_t and $reset_t$ are the update and reset gate vectors, respectively, and $\tilde{h}_t$ is the candidate hidden state. Compared to traditional Recurrent Neural Networks, GRUs effectively mitigate the vanishing gradient problem with their gating mechanisms. GRUs achieve similar performance to LSTM networks but with a simpler architecture and fewer parameters, leading to faster training and reduced computational complexity. Therefore, GRUs are ideal for predicting drone movement trends from RSSI data.

3) Hidden layer: The hidden layer is composed of several neurons that further process the data from the GRU layer, extracting more complex features.

4) Exponential layer: An "Exponential Layer" in a neural network serves as a specialized activation function designed to transform each input value x into 10^x. In drone-GCS communications, signal strength exhibits an exponential relationship with distance, as described by the LDPL, with minimal interference from multipath and reflections in line-of-sight (LOS) channels. Traditional activation functions, however, struggle to capture this relationship efficiently, leading to slow training and poor convergence. To address this, the R2D-GRU network incorporates the Exponential Layer.

5) Output layer: The output layer produces the normalized distance data ($\hat{D}'_t$), generating the predicted normalized distance series ($\hat{\mathbf{D}}'_{t_c}$).

This network is lightweight, has a high recognition rate, and can handle variable-length sequential data inputs, making it suitable for real-time spoofing attack detection tasks on drones.

D. IQR-based detection

After obtaining the predicted normalized distance variation trend, we assess whether the spoofing attack has occurred by measuring its Pearson correlation with the actual normalized distance variation trend. The Pearson correlation coefficient, r_{t_c}, is calculated using the formula:

$$r_{t_c} = \frac{\sum_{t=1}^{T}(D'_t - \mu_{D'})(\hat{D}'_t - \mu_{\hat{D}'})}{\sqrt{\sum_{t=1}^{T}(D'_t - \mu_{D'})^2}\sqrt{\sum_{t=1}^{T}(\hat{D}'_t - \mu_{\hat{D}'})^2}}, \tag{10}$$

where D'_t and $\hat{D}'_t$ are the actual and predicted normalized distance values at time t, respectively. $\mu_{D'}$ and $\mu_{\hat{D}'}$ are the mean values of the actual and predicted normalized distance values, respectively, and T is the number of observations.

To determine the threshold θ for spoofing attack detection, we use the Interquartile Range (IQR) method, which is a classic method from the field of anomaly detection [25], [26]. The IQR is a measure of statistical dispersion and is calculated as the difference between the 75th percentile (Q3) and the 25th percentile (Q1) of the Pearson correlation coefficients. The threshold is then set using the following formula:

$$\theta = Q1 - k \times \text{IQR}, \tag{11}$$

$$\text{IQR} = Q3 - Q1, \tag{12}$$

where k is a constant factor typically set to 1.5, which is also used in our work. Specifically, we calculate the threshold using the training set that contains only positive samples (normal samples without spoofing attacks) and use it to evaluate the testing data that contains both positive samples and negative samples (samples with spoofing attacks). This approach ensures generalizability, as the threshold is based solely on positive samples, allowing any deviation to indicate a spoofing attack. Additionally, in some extreme cases, the value of θ may be less than 0. To ensure that the actual normalized distance values and the predicted normalized distance values maintain a positive correlation, we set the threshold to 0.

Finally, by comparing the Pearson correlation coefficient of the time series with the IQR-based threshold, we determine whether the drone is under a spoofing attack at time t_c. If the Pearson correlation coefficient is greater than or equal to the threshold, we classify the situation as normal. If it is less than the threshold, we consider the drone-GCS communication to be under a spoofing attack.

$$\text{Status} = \begin{cases} \text{Normal}, & \text{if } r_{t_c} \geq \theta; \\ \text{Under Spoofing Attack}, & \text{if } r_{t_c} < \theta. \end{cases} \tag{13}$$

Overall, DroneMA addresses the challenge of predicting distance from RSSI by effectively utilizing temporal RSSI information to infer drone movement trends through normalization preprocessing and a deep learning-based approach. The IQR-based threshold method, which uses only positive samples to define the classification boundary, can resist various spoofing attacks, including those not present in the negative samples. The network structure used in this scheme is simple, with few parameters, and the computational complexity for detection remains polynomial, making it suitable for deployment on resource-constrained drones.

V. Experimental Evaluation

Currently, there is a lack of publicly available datasets that provide RSSI and distance data for drones, and simulated data often fail to represent real-world conditions accurately. In this section, we have conducted real-world experiments to evaluate the performance of proposed schemes.

A. Experimental setup

1) Experimental device configuration: In this subsection, we provide a detailed description of the drone and GCS settings used in the experiment.

As illustrated in Fig. 4, the experimental drone is a quadcopter equipped with a Pixhawk 6c mini flight controller and an NVIDIA Jetson Nano B01 4GB as the onboard computer. In our experiments, we utilized only some of the sensors, including the IMU sensors integrated with the Pixhawk 6c mini and a Cirocomm PA025AZ0009 GPS receiver. Additionally, the telemetry radio employed is one of the most popular radio platforms for drones, the SiK Telemetry Radio V3. This telemetry radio operates at a frequency of 915 MHz with a maximum output power of 500 mW. It provides an RSSI value in the range of 0-255, which can be converted to received signal strength (RSS) in dBm using the following formula:

$$\text{RSS (dBm)} = \left(\frac{\text{RSSI}}{1.9}\right) - 127. \tag{14}$$

On the GCS, we used an HP laptop equipped with the Mission Planner application. This laptop was connected to a SiK Telemetry Radio V3, the same radio used on the drone.

During the experiment, the telemetry radio was used to connect the drone and the GCS. They communicated via the MAVLink protocol, a lightweight messaging protocol for drones, which transmitted real-time status information of the drone. To avoid the influence of takeoff and landing phases, we only collected data when the drone was at a fixed altitude.

Specifically, we collected three types of MAVLink messages: RADIO_STATUS, LOCAL_POSITION_NED, and MAVLINK_HOME_POSITION. These messages were set to a transmission frequency of 1Hz to continuously collect the current RSSI information, the current position information, and the home position information (used to approximate the distance between the drone and the GCS).

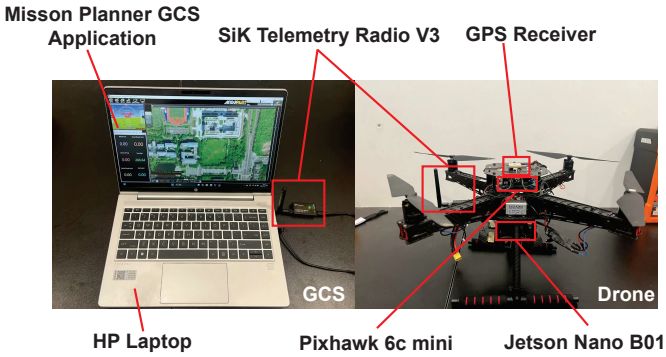


Fig. 4: Experimental setup for drone and GCS.

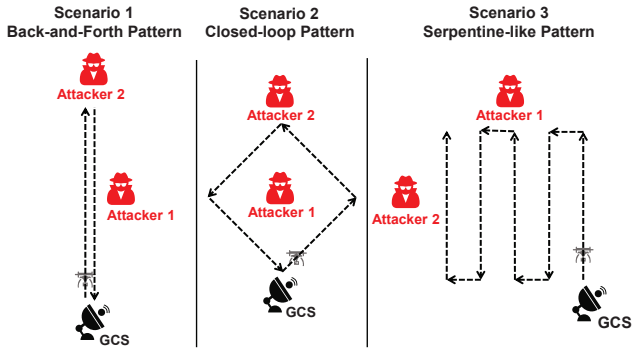


Fig. 5: Experimental scenarios corresponding to 3 common drone movement patterns.

2) Scenario configuration: To design realistic and effective experiments, we devised three experimental scenarios based on common drone tasks:

1) **Back-and-Forth pattern:** This pattern involves the drone moving along a straight path back and forth, which is often used for monitoring linear or elongated terrains, as well as for cargo transportation between two points.
2) **Closed-loop pattern:** In this pattern, the drone follows a closed loop, aimed at covering and monitoring an enclosed area, which is commonly used for area patrol, 3D modeling, and multi-point cargo transportation.
3) **Serpentine-like pattern:** The drone executes a motion in this pattern similar to a serpentine-like path, designed to systematically cover large areas, which is often used for agricultural monitoring, land surveying and scanning.

To implement these three scenarios, the GCS was positioned near the drone's takeoff location, with two attackers placed midway and at the farthest point from the GCS along the drone's flight path. Data communicated with the GCS were considered positive samples, while data from the attackers were considered negative samples. In Scenarios 1 and 2, the drone repeated the motion three times, while in Scenario 3, the drone executed the motion once. Fig. 7 illustrates the flight mission. For each scenario, we collected 10 sets of flight data, and for each attack condition, we collected 5 sets, totaling 60 sets. This yielded 7,312 positive samples and 6,887 negative samples. The sample distribution is summarized in Table I.

3) R2D-GRU network configuration: The R2D-GRU network was implemented using PyTorch. The specific network

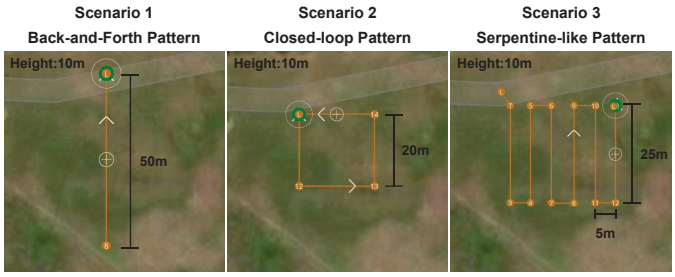


Fig. 6: Flight mission executed by the drone. Scenario 1: 3 back-and-forth runs (300m total). Scenario 2: 3 loops (240m total). Scenario 3: 1 run (175m total). Due to security considerations, our experiments were conducted within a limited range. Nonetheless, we meticulously accounted for various relative positions among the drone, attacker, and GCS, thereby ensuring that our findings are comparable to those obtained from experiments conducted over greater distances.

TABLE I: Distribution of positive and negative samples across experimental scenarios.

Experimental Scenario	Positive Samples	Negative Samples
Scenario 1	2737	2678
Scenario 2	2698	2343
Scenario 3	1877	1866
Total	7312	6887

parameters are summarized in Table II. The network has only 4,253 trainable parameters, demonstrating its lightweight nature. This ensures rapid training and fast inference on resource-constrained drone terminals, thereby guaranteeing the real-time performance of the detection scheme. During training, we used only positive samples to train the network. The training process consisted of 100 epochs with a learning rate of 0.001, using the Adam optimizer and MSE loss function.

TABLE II: Model summary of R2D-GRU network.

Layer (Type)	Input Shape	Output Shape	# Param
GRU Layer (GRU)	1	32	3264
Hidden Layer (Fully connected)	32	32	1056
Exponential Layer (Activation)	32	32	0
Output Layer (Fully connected)	32	1	33
Total params: 4253			

B. Effectiveness in spoofing attack detection

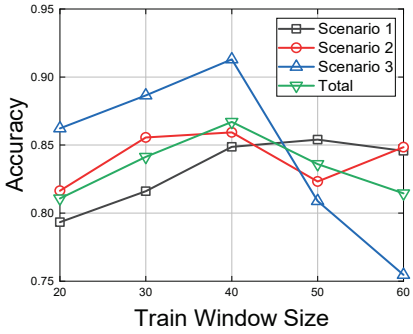
To evaluate the effectiveness of the spoofing attack detection framework, we treat the detection task as a binary classification problem. We use the standard performance metrics: Accuracy, Precision, Recall, and F1 Score.

1) Optimal train window size for DroneMA: In this section, we analyze the impact of different train window sizes on the performance of the DroneMA. As illustrated in Figure 7a, the accuracy increases with the size of the train window to 40, after which it starts to decrease in all scenarios. The likely explanation for this trend is that when the window size is less than 40, the period is too short for the network to adequately capture the correlation between RSSI and distance. On the other hand, when the window size exceeds 40, the drone's position and environment may undergo substantial changes over an extended period. Consequently, information collected earlier in the window may become irrelevant to the present time, leading to a reduction in the detection efficiency. Based on these results, a train window size of 40 appears to be the optimal choice. This window size captures sufficient temporal dynamics to enhance the model's ability to detect spoofing attacks while avoiding the pitfalls of overfitting.

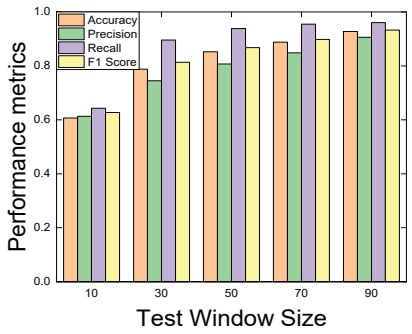
2) Detection performance under different test window sizes: DroneMA is capable of accepting variable-length time series for analysis. However, at the initial stages of a flight mission, the length of the selectable time window is constrained by the limited data available. Only after a certain duration of flight can longer time windows be effectively utilized for more comprehensive analysis. Consequently, it is imperative to explore the detection performance of DroneMA under varying time window sizes to understand how the duration of the time window impacts the accuracy of the spoofing attack detection. In Fig. 7b, the detection performance is evaluated using four metrics: Accuracy, Precision, Recall, and F1 Score, across different test window sizes for three scenarios (Scenario 1, Scenario 2, and Scenario 3) as well as the combined scenario (Total). As the test window size increases, all metrics show steady improvement, indicating that longer test windows lead to better spoofing attack detection performance. The detection accuracy for each scenario with the test window size set to 90 is 95.85%, 87.62%, and 96.43%, respectively. Overall, the combined scenario achieves a detection accuracy of 92.78%. DroneMA's Recall is consistently high, indicating the model's effectiveness in recognizing normal conditions, which is beneficial for the drone's regular operation as frequent false negatives would significantly impact its usability. The Precision is slightly lower, suggesting some false positives. The close values of F1 Score and Accuracy imply that the model is well-balanced in distinguishing between normal and spoofed scenarios and that the dataset has a reasonable proportion of positive and negative samples.

3) Comparative Analysis of DroneMA with Traditional Methods: To demonstrate the superiority of our proposed scheme, we compared DroneMA with traditional methods based on the LDPL and LDPL with Gaussian filtering [27], [28]. In these comparative methods, the Z-score normalization and R2D-Network components of our framework were replaced with traditional LDPL-based methods, as shown in Equation 1. The RSSI_0 value was determined through direct measurement, while the path loss exponent n was estimated using the least squares method, resulting in a value of 2.4.

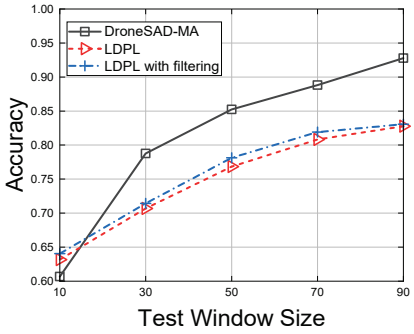
As depicted in Fig. 7c, in the combined scenarios considering all three test scenarios, when the test window size was 10, the LDPL method slightly outperformed our approach. This is because, over short periods, the DroneMA scheme may not capture the communication environment characteristics accurately. However, at other window lengths, the LDPL-based methods consistently showed lower accuracy. Traditional wireless ranging methods fail to effectively reflect drone



(a) Accuracy of DroneMA for different train window sizes in individual scenarios and combined scenarios. Test window size is 60.



(b) Performance under different test window sizes considering the combined scenarios. The train window size is 40.



(c) Comparison of DroneMA and traditional LDPL methods (with and without filtering) across different test window sizes.

Fig. 7: Evaluation results of DroneMA.

mobility, resulting in poor differentiation between normal and spoofed scenarios. As a result, when calculating the IQR threshold, the values consistently fell below zero. Additionally, traditional distance estimation methods rely solely on the current RSSI value, disregarding previous data. Since RSSI values can remain constant over some time, the distance calculated using path loss-based RSSI methods also remains unchanged, leading to identical series that cannot compute the Pearson correlation coefficient, rendering spoofing attack detection ineffective during these periods. Introducing filtering marginally improved the accuracy of the LDPL method. While filtering can reduce noise, it also diminishes the ability to respond to drone mobility. Thus, filtering methods may work in static ranging environments but struggle in dynamic scenarios. Notably, we also experimented with incorporating Gaussian and Kalman filtering in DroneMA but observed a slight performance decline. We hypothesize this is because filtering reduces the details in RSSI reflecting drone movements. Additionally, our GRU-based prediction network inherently possesses denoising capabilities due to its ability to remember past information, capturing hidden drone movement information in the RSSI series without needing explicit filtering. In summary, our method adapts to drone mobility over slightly longer flight times, providing more accurate spoofing attack detection compared to traditional methods.

4) Efficiency of DroneMA: To illustrate the lightweight nature of our proposed DroneMA scheme, we deployed the algorithm on the drone's onboard computer and measured the processing time and floating-point operations (FLOPs) required for different test window sizes. Table III shows the distribution of processing time and FLOPs across test window sizes ranging from 10 to 90. The results demonstrate that even for the largest test window size, the processing time remains under 12 milliseconds, and the FLOPs count stays below 800,000. These values highlight the efficiency and computational feasibility of DroneMA, making it suitable for deployment on resource-constrained drone nodes. The low computational overhead ensures that the scheme can operate in real-time, providing timely and reliable spoofing attack detection without significantly draining the drone's resources.

TABLE III: Processing time and FLOPs of DroneMA across different test window sizes.

Test window size	10	30	50	70	90
Processing time(ms)	3.075	5.532	7.341	9.972	11.802
FLOPs	87824	263464	439104	613764	790384

VI. Conclusion

In this paper, we delved into the detection of spoofing attacks within drone-GCS networks. By harnessing the concept of mobility alignment, which is reflected in both the communication and sensing systems of drones, the proposed DroneMA framework adeptly accommodates the high mobility and dynamic environmental conditions characteristic of drone scenarios, facilitating real-time, continuous spoofing attack detection. Experimental results demonstrated that the DroneMA framework delivers commendable performance across three tested scenarios, achieving a detection accuracy of 92.78% with a test window size of 90. However, to ensure that our solution is not only effective but also resilient under a wide array of conditions, future work will focus on enhancing the robustness of the system. Several challenges need to be addressed, including the formal analysis of the proposed solution, the application of longer and more distant flight missions in testing, the incorporation of a broader range of attacker strategies (such as dynamic attacks and multi-attacker scenarios), and the consideration of drone swarm scenarios. By addressing these issues, DroneMA will not only maintain its effectiveness but also exhibit greater robustness in the face of diverse security conditions.

Acknowledgment

This work was supported in part by the National Key R&D Program of China under Grant 2022YFB3103500, in part by the National Natural Science Foundation of China under Grant 62101079, in part by the Venture and Innovation Support Program for Chongqing Overseas Returnees under Grant cx2021012.

REFERENCES

[1] P. G. Fahlstrom, T. J. Gleason, and M. H. Sadraey, *Introduction to UAV systems*. John Wiley & Sons, 2022.

[2] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in uav communication networks," *IEEE communications surveys & tutorials*, vol. 18, no. 2, pp. 1123–1152, 2015.

[3] Z. Wang, Y. Li, S. Wu, Y. Zhou, L. Yang, Y. Xu, T. Zhang, and Q. Pan, "A survey on cybersecurity attacks and defenses for unmanned aerial systems," *Journal of Systems Architecture*, vol. 138, p. 102870, 2023.

[4] B. Nassi, R. Bitton, R. Masuoka, A. Shabtai, and Y. Elovici, "Sok: Security and privacy in the age of commercial drones," in *2021 IEEE symposium on security and privacy (SP)*. IEEE, 2021, pp. 1434–1451.

[5] J. Sharma and P. S. Mehra, "Secure communication in iot-based uav networks: A systematic survey," *Internet of Things*, p. 100883, 2023.

[6] Y. Shi, K. Davaslioglu, and Y. E. Sagduyu, "Generative adversarial network in the air: Deep adversarial learning for wireless signal spoofing," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 1, pp. 294–303, 2020.

[7] ——, "Generative adversarial network for wireless signal spoofing," in *Proceedings of the ACM Workshop on Wireless Security and Machine Learning*, 2019, pp. 55–60.

[8] R. Aissaoui, J.-C. Deneuville, C. Guerber, and A. Pirovano, "A survey on cryptographic methods to secure communications for uav traffic management," *Vehicular Communications*, p. 100661, 2023.

[9] K. Zeng, K. Govindan, and P. Mohapatra, "Non-cryptographic authentication and identification in wireless networks [security and privacy in emerging wireless networks]," *IEEE Wireless Communications*, vol. 17, no. 5, pp. 56–62, 2010.

[10] N. Wang, L. Jiao, P. Wang, W. Li, and K. Zeng, "Machine learning-based spoofing attack detection in mmwave 60ghz ieee 802.11 ad networks," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 2579–2588.

[11] H. Liu, Y. Wang, J. Liu, J. Yang, and Y. Chen, "Practical user authentication leveraging channel state information (csi)," in *Proceedings of the 9th ACM symposium on Information, computer and communications security*, 2014, pp. 389–400.

[12] J. Yang, Y. Chen, W. Trappe, and J. Cheng, "Detection and localization of multiple spoofing attackers in wireless networks," *IEEE Transactions on Parallel and Distributed systems*, vol. 24, no. 1, pp. 44–58, 2012.

[13] L. Xiao, X. Wan, and Z. Han, "Phy-layer authentication with multiple landmarks with reduced overhead," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 1676–1687, 2017.

[14] W. Hou, X. Wang, and J.-Y. Chouinard, "Physical layer authentication in ofdm systems based on hypothesis testing of cfo estimates," in *2012 IEEE International Conference on Communications (ICC)*. IEEE, 2012, pp. 3559–3563.

[15] K. Sankhe, M. Belgiovine, F. Zhou, S. Riyaz, S. Ioannidis, and K. Chowdhury, "Oracle: Optimized radio classification through convolutional neural networks," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 370–378.

[16] S. Jana and S. K. Kasera, "On fast and accurate detection of unauthorized wireless access points using clock skews," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, 2008, pp. 104–115.

[17] C. Pei, N. Zhang, X. S. Shen, and J. W. Mark, "Channel-based physical layer authentication," in *2014 IEEE Global Communications Conference*. IEEE, 2014, pp. 4114–4119.

[18] L. Bai, L. Zhu, J. Liu, J. Choi, and W. Zhang, "Physical layer authentication in wireless communication networks: A survey," *Journal of Communications and Information Networks*, vol. 5, no. 3, pp. 237–264, 2020.

[19] T. Alladi, G. Bansal, V. Chamola, M. Guizani *et al.*, "Secauthuav: A novel authentication scheme for uav-ground station and uav-uav communication," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 15 068–15 077, 2020.

[20] N. Wang, J. Duan, B. Chen, S. Guo, T. Xiang, and K. Zeng, "Efficient group key generation based on satellite cluster state information for drone swarm," *IEEE Transactions on Information Forensics and Security*, 2024.

[21] Y. Zhou, P. L. Yeoh, K. J. Kim, Z. Ma, Y. Li, and B. Vucetic, "Game theoretic physical layer authentication for spoofing detection in uav communications," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 6, pp. 6750–6755, 2022.

[22] Y. Zhou, Z. Ma, H. Liu, P. L. Yeoh, Y. Li, B. Vucetic, and P. Fan, "A uav-aided physical layer authentication based on channel characteristics and geographical locations," *IEEE Transactions on Vehicular Technology*, 2023.

[23] Y. Teng, P. Zhang, X. Chen, X. Jiang, and F. Xiao, "Phy-layer authentication exploiting channel sparsity in mmwave mimo uav-ground systems," *IEEE Transactions on Information Forensics and Security*, 2024.

[24] L. Xiao, L. J. Greenstein, N. B. Mandayam, and W. Trappe, "Channel-based detection of sybil attacks in wireless networks," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 3, pp. 492–503, 2009.

[25] H. Vinutha, B. Poornima, and B. Sagar, "Detection of outliers using interquartile range technique from intrusion dataset," in *Information and decision sciences: Proceedings of the 6th international conference on ficta*. Springer, 2018, pp. 511–518.

[26] P. J. Rousseeuw and M. Hubert, "Anomaly detection by robust statistics," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 2, p. e1236, 2018.

[27] P. Wang and Y. Luo, "Research on wifi indoor location algorithm based on rssi ranging," in *2017 4th International Conference on Information Science and Control Engineering (ICISCE)*. IEEE, 2017, pp. 1694–1698.

[28] Y. Huang, J. Zheng, Y. Xiao, and M. Peng, "Robust localization algorithm based on the rssi ranging scope," *International Journal of Distributed Sensor Networks*, vol. 11, no. 2, p. 587318, 2015.