

# Deceptive Skies: Leveraging GANs for Drone Sensor Data Falsification

Mehmed Kerem Uludag<sup>1</sup>, Maryna Veksler<sup>2</sup>, Yasin Yilmaz<sup>3</sup>, Kemal Akkaya<sup>2</sup>

<sup>1</sup>Department of Computer Science and Robotics, <sup>2</sup>Knight Foundation School of Computing and Information Sciences,

<sup>3</sup>Department of Electrical Engineering

<sup>1</sup>University of Michigan, <sup>2</sup>Florida International University, <sup>3</sup>University of South Florida

muludag@umich.edu, mveks001@fiu.edu, yasiny@usf.edu, kakkaya@fiu.edu

## ABSTRACT

Drones, embodying the spirit of innovation, have become dynamic game-changers, redefining industries through their unparalleled adaptability and budget-friendly solutions. However, compromised drone sensors pose serious safety risks. Therefore, multiple studies have been dedicated to analyzing various drone sensor attacks and developing efficient anomaly detectors.

In this paper, we utilize GANs to design a novel model named DS-GAN for drone sensor data falsification that can be used for false data injection (FDI) attacks. Furthermore, considering the presence of multiple sensors on a drone, we analyze both their temporal and spatial relationships to develop an ensemble DSD-GAN (EDSD-GAN), designed for executing more sophisticated yet subtle attacks. We evaluate the proposed FDI attacks against two popular machine learning (ML) architectures used for drone sensor anomaly detection. Our extensive experiments demonstrate that the proposed attack is highly effective for various drone sensors with an average success rate above 80%. Moreover, we demonstrate that using EDSD-GAN significantly improves the attack success rate by over 50% even for the most complex cases.

## CCS CONCEPTS

• **Computer systems organization** → *Sensors and actuators*; • **Security and privacy** → *Intrusion detection systems*; • **Computing methodologies** → *Machine learning*.

## KEYWORDS

generative adversarial networks (GANs), drones, anomaly detection

### ACM Reference Format:

Mehmed Kerem Uludag<sup>1</sup>, Maryna Veksler<sup>2</sup>, Yasin Yilmaz<sup>3</sup>, Kemal Akkaya<sup>2</sup>. 2024. Deceptive Skies: Leveraging GANs for Drone Sensor Data Falsification. In *The 39th ACM/SIGAPP Symposium on Applied Computing (SAC '24)*, April 8–12, 2024, Avila, Spain. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3605098.3636059>

## 1 INTRODUCTION

Drones, or Unmanned Aerial Vehicles (UAVs), have transformed industries like surveillance, agriculture, and disaster management

due to their cost-effectiveness. However, compromised sensors can lead to malfunctions and pose safety risks [2, 5]. Studying threats against drone sensors is crucial for developing effective security systems.

Extensive research has focused on anomaly detection for drone sensors. Earlier works utilized Extended Kalman Filter (EKF) as the primary intrusion detection system (IDS) [7, 9]. Recent advancements in machine learning (ML) led to more complex systems, which can be categorized based on their operations, such as learning benign behavior [11, 15, 19] or distinguishing between benign and specific anomalies like spoofing or jamming [6]. The commonality is their emphasis on inertial measurement unit (IMU) drone sensors for effective anomaly detection.

While there is a lot of research devoted to drone sensor anomaly detectors, the existing methods are mostly based on training with limited data or data specific to a given attack. Traditional attacks involve manual falsification, such as GPS spoofing and jamming using Software Defined Radio (SDR) devices [2, 3]. Furthermore, with false data injection (FDI) attacks, an attacker may directly inject false readings into sensors [4]. However, the advancement of generative adversarial networks (GANs) introduces possibilities for more sophisticated falsified sensor data through false data injection (FDI) attacks.

In this paper, we introduce DSD-GAN, a GAN architecture for the falsification of drone sensor data. Our model enables the generation of high-quality fake data for individual accelerometer and gyroscope IMU sensor readings, facilitating the use of GANs in FDI attacks against drone IMU sensors. To address spatial relationship challenges, we design an ensemble of DSD-GANs, EDSD-GAN, combining individual sensor models and a combined meta-model. EDSD-GAN facilitates falsifying data based on multi-sensor relationships, making it challenging for traditional detectors to flag.

We evaluate our GAN-based FDI attack against two state-of-the-art anomaly detectors representing the most commonly used ML architectures, convolutional neural networks (CNNs), and autoencoders. We conduct an extensive set of experiments to investigate the relationship between the attack success (i.e., the rate of success at deceiving the classifier) and the number of sensors affected by the attack. Furthermore, we compare the performance of multi-sensor FDI attacks executed using individual DSD-GANs and EDSD-GAN. Our results demonstrate the effectiveness of both proposed attacks in deceiving existing drone sensor anomaly detectors, with an attack success rate exceeding 80% for given drone sensors. Notably, using EDSD-GAN improves the attack success rate by at least twofold compared to DSD-GAN.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

SAC '24, April 08–12, 2024, Spain

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00

<https://doi.org/10.1145/3605098.3636059>

The paper is organized as follows: Section 2 reviews related work, Section 3 outlines the system and threat model, Section 4 presents the attack methodology, and Section 5 evaluates the proposed attack. Finally, Section 6 concludes the paper.

## 2 RELATED WORK

In this section, we discuss the relevant work on drone sensor attacks, sensor-based anomaly detectors for drones, and generative adversarial networks (GANs) used for adversarial ML.

**Drone Sensor Attacks:** The majority of drones come equipped with a diverse array of sensors to ensure their proper functioning. These sensor readings play a crucial role in providing essential information about the drone's state and position to its Ground Control Station (GCS), influencing control decisions. Given the pivotal role of sensors in drone operations, attackers have devised various types of attacks directly impacting these sensors, including GPS spoofing, jamming, and FDI.

GPS spoofing attacks involve the falsification of GPS signals with the intent of disrupting a drone's flight by manipulating its sensors. In this attack, an adversary injects false GPS signals into the shared medium, leading the drone to perceive them as legitimate, thereby compromising the mission's success. Notably, in [2], the authors executed a GPS spoofing attack against a 3DR Solo drone using SDR, resulting in complete drone hijacking. Similarly, Saputo et al. [3] employed an SDR device to demonstrate the effectiveness of a GPS spoofing attack against a DJI Phantom 3 drone. Jamming attacks, on the other hand, entail an attacker compromising the shared communication medium. Unlike spoofing attacks, the primary aim of jamming attacks is to disrupt communication between the drone and its GCS and forcibly manipulate the drone's flight mode. In one instance, authors in [13] implemented uninterrupted radio jamming to disrupt the operations of a rogue drone. FDI attacks are more intrusive and targeted compared to spoofing and jamming attacks. They involve an adversary injecting false sensor readings directly into the drone system. In [4], a Kalman Filter-based FDI attack was developed to manipulate drone position control. Chen et al. [5] employed a jamming approach to inject false magnetometer readings into a victim drone, successfully causing the drone to crash or alter its trajectory.

Differently to previous approaches that manually falsify sensor readings for drone attacks, our work utilizes Generative Adversarial Networks (GANs) to automate the sensor data generation process. Furthermore, we generate data for different sensors concurrently to preserve their original spatial relationships.

**Drone Anomaly Detectors:** Given the multitude of attacks that can compromise drone operations through its sensors, substantial effort has been dedicated to designing effective methods for sensor anomaly detection. In [11], the authors proposed ML-based method to detect drone sensor spoofing. They employed a multi-layer perceptron (MLP) trained to predict the next position of the aircraft. This system identifies anomalies when the prediction error exceeds 1 meter, with a particular focus on inertial measurement units (IMU) sensors. In [19], researchers developed a one-class detection technique for drone sensor anomalies. Their evaluation of various ML-based classifiers highlighted that the autoencoder neural network achieved the highest average F1 score of 94.81%, tested

across various drone models. Galvan et al. [6] devised a Convolutional Neural Network (CNN) model to identify anomalous sensor faults by analyzing IMU sensor readings. The work presented in [15] proposed the utilization of a modified Long-Short Term Memory (LSTM) model for anomaly detection using drone sensor data and state information. Trained on benign data, this model learns to predict the future state of the drone system and sensors based on historical data. When tested against various manually injected faults across IMU sensors, the system achieved an average F1 score exceeding 95%.

Our research reveals that various anomaly detectors are vulnerable to GAN-based FDI attacks. Additionally, we identify that GAN-generated sensor data can be leveraged to analyze ML classifiers and extract valuable insights into their decision-making processes.

**GANs and Adversarial ML:** Introduced in 2014 [8], generative adversarial networks (GANs) are often employed to generate realistic fake data. GANs are frequently used to create deep fakes and can be utilized to generate synthetic data of various types, either from scratch or with additional modifications, leading to an increase in the number of attacks using GANs. In [17], the authors used GANs to evade machine learning-based IDS for network traffic. They subsequently demonstrated that GAN-generated data can also be used to improve IDS performance by training on adversarial perturbations. Authors in [12] developed a bi-objective GAN to mislead Android malware detection systems. Shi et al. [16] designed a spoofing attack based on GANs, successfully generating high-quality synthetic signals for conducting a spoofing attack with a 76.2% success probability. In [10], GANs were employed to deceive radio frequency (RF)-based authentication mechanisms, achieving over a 90% success rate when attempting to fool the authentication devices.

In contrast to the aforementioned approaches, we propose the use of an ensemble of GANs to generate high-quality sensor data that captures their temporal and spatial correlations. Additionally, our ensemble includes a state discriminator to ensure that the falsified data accurately reflects a specific drone's state.

## 3 SYSTEM MODEL AND ADVERSARY MODEL

In this section, we present the control system model and threat model considered in this paper.

### 3.1 Control System Model

The control system model is composed of a drone and a benign GCS, engaging in communication with each other, as depicted in Figure 1.

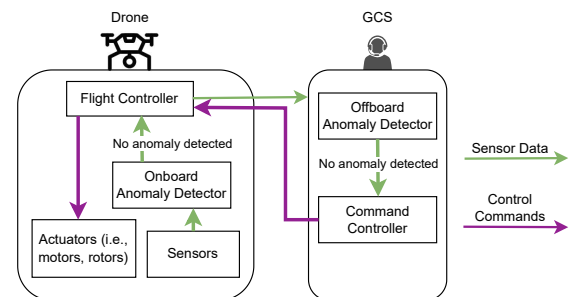


Figure 1: Control System Model.

We consider two key modules of the drone as follows:

**Drone Sensors:** A typical drone is equipped with an array of sensors, including GPS, accelerometers, tilt sensors, and an inertial measurement unit (IMU). These sensors play a vital role in determining the drone’s position, orientation, and component status, which are crucial for effective aircraft control.

**Flight Controller (FC):** In this work, the Flight Controller (FC) collects sensor data from the aircraft, serving two primary functions: Transmitting sensor data to the GCS and directly triggering actions of actuators based on received sensor readings. The GCS station, upon receiving sensor data, sends control messages to the FC, which in turn activates the actuators.

The GCS station comprises two primary components:

**Sensor Anomaly Detector:** The proposed system can incorporate up to two anomaly detectors, installed on-board and off-board. These detectors serve as a protective mechanism designed to identify abnormal drone behaviors based on locally received or FC-transmitted sensor data. The anomaly detector is equipped with a pre-trained ML algorithm specialized in detecting spoofing and jamming attacks on the drone. Furthermore, the model continues to learn from sensor readings in real-time during drone flight.

**Command Controller:** Once the anomaly detector validates the sensor data as benign, the Command Controller takes action. It utilizes this verified data to formulate and transmit control messages to the drone, specifying the desired trajectory for the aircraft.

### 3.2 Threat Model

In this work, we consider an adversary with the capability to both observe the actual sensor readings of the drone and transmit modified sensor data to the GCS.

During the observation phase, the adversary collects benign sensor data, which serves as a reference for creating fabricated data. Alternatively, the adversary can obtain benign drone sensor data from open-source datasets or by acquiring and flying the same model of drone.

Subsequently, the falsified data can be transmitted by either 1) modifying the sensor data as it exits the drone using a Man in the Middle (MiTM) attack. This may involve blocking the drone’s traffic, impersonating it, and sending altered sensor readings; or 2) manipulating the sensors of the drone directly, such as spoofing GPS sensor data. Additionally, 3) completely compromising the drone, potentially through malware or modified firmware, can also be used as a means to transmit falsified data. Figure 2 provides a visual representation of all these scenarios.

First, an adversary can adopt a MiTM approach, positioning between the drone and its GCS. Initially, the adversary intercepts and blocks communication signals carrying onboard sensor data and system measurements from the drone to the GCS. Subsequently, the adversary sends custom/modified sensor data to the GCS, portraying a false reality of the drone’s state. This manipulation leads the GCS to issue control commands that could potentially trigger unpredictable and hazardous behaviors in the drone.

Secondly, an adversary can transmit malicious signals to the drone through a GPS spoofing attack to modify its sensors’ data. Thus, falsified sensor GPS data is sent to the drone to disrupt its normal operations.

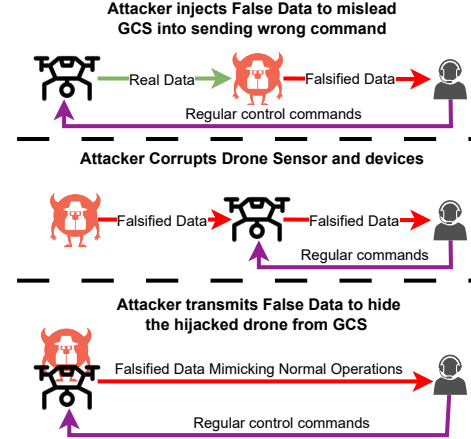


Figure 2: Threat Model.

A third technique consists of injecting sensor data directly into the drone to either 1) conceal the anomalous behavior of a drone that has already been hijacked or 2) trick the GCS into sending the control commands based on the incorrect data. In these scenarios, an adversary is capable of directly injecting falsified sensor data onto the drone FC. For cases when the drone is hijacked and the main goal of an adversary is to mimic normal drone operations, the falsified sensor data is crafted to reflect the reaction to the regular control commands received from the benign GCS, while in reality, the drone is under an attacker’s control. Therefore, an adversary gains complete control over the compromised drone, while remaining undetected by the GCS. Alternatively, an adversary can inject falsified sensor data with the goal of tricking the GCS into sending malicious commands based on inaccurate information about the state of the drone.

## 4 PROPOSED DECEPTION METHODOLOGY

Given the attack model, in this section we propose to utilize GANs for the execution of FDI attack. To this end, we follow a three-step process depicted in Figure 3.

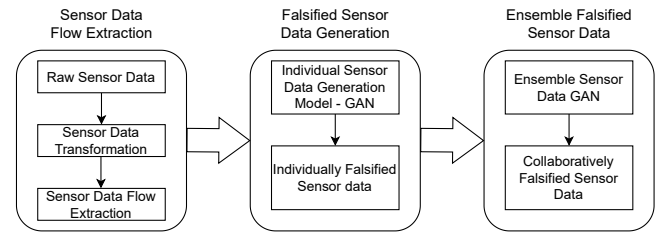


Figure 3: The proposed attack scheme.

### 4.1 Overview of the Attack Setup

The main goal of an attacker is to fabricate drone sensor data indistinguishable from real sensor readings in the eyes of a designated sensor anomaly detector. Due to its ability to generate high-quality data [1], we select GAN as a primary tool to produce adversarial samples. Moreover, using GANs allows us to capture specific features directly from the benign samples instead of reproducing

them from scratch. The key stages of the proposed attack shown in Figure 3 can be outlined as follows:

**(1) Sensor Data Flow Extraction.** We gather raw data from  $M$  sensors simultaneously active on the drone and convert them into time-series vectors. Specifically, we extract  $N$  sliding windows of size  $W$  with a stride of  $s$ , creating sequences of events for each sensor denoted as  $S_{rl} = \{S_{rl}^1, S_{rl}^2, \dots, S_{rl}^N\}$ .

**(2) Fabricated Sensor Data Generation.** Utilizing the acquired sensor data vectors, we train  $M$  distinct sensor data generation models denoted as  $G = \{G_1, G_2, \dots, G_M\}$ .

**(3) Ensemble of Falsified Sensor Data.** An aggregator generator model  $E_G$  is applied to ensemble all individual  $S_{fake}$  vectors in a collaborative manner.

We use the proposed scheme to execute multiple FDI attacks on the drone IMU sensors. We design minimal and incremental intrusion FDI attacks, by gradually increasing the number of compromised IMU sensors. We assume that two distinct ML-based anomaly detectors are employed by the drone control system as a protection mechanism.

## 4.2 Model Design

In this section, we develop a novel sensor data generator model, namely DSD-GAN, to produce realistic readings for various drone sensors in desired spatial and temporal domains.

As indicated in Figure 4, DSD-GAN consists of 2 deep learning models,  $G$  (Generator) and  $D$  (Discriminator). Specifically,  $G$  uses latent space of samples,  $X$ , to produce a set of falsified readings  $G(X)$ .  $D$  computes the difference between  $G(X)$  and  $S_{rl}$  to evaluate the authenticity of the falsified sensor data. The details of each model are as follows:

**Control Parameter (z\_dim).** A user-defined value for  $z\_dim$  also determines the sporadic nature of the data. This parameter initializes an architectural dimension of  $G$  and thus, is used to determine the width of the latent sample input for  $G$ .

**Generator ( $G$ ).** In our application,  $G$  plays a crucial role in generating a vector of falsified sensor data that needs to be virtually indistinguishable from real sensor readings. We use Linear layers to serve as an initial mapping from a latent space to a higher-dimensional feature space. Linear layers are valuable for their simplicity and their capacity to linearly transform the input data. This initial transformation helps set the foundation for subsequent layers to capture more complex relationships.

To account for the temporal and spatial dependencies within the drone sensor data, we incorporate 1D Convolutional (Conv 1D) layers into our architecture. Conv 1D layers are well-suited for detecting patterns and features in sequential data, making them ideal for capturing the time-varying aspects of sensor readings within our desired window.

Additionally, we leverage Long Short-Term Memory (LSTM) layers, which are a type of recurrent neural network, to model and capture long-range dependencies in the sensor data. Our use of LSTM layers allows us to capture more extended, apparent patterns across windows, which the Conv 1D layers may lack. This is particularly important, as drone sensor data, such as altimeter and accelerometer readings, often exhibits intricate patterns and correlations over time.

Furthermore, between the Conv 1D and LSTM layers, we employ a series of reshaping operations. These reshaping operations are essential for handling the differences in data channel dimensions. They not only ensure that the data is properly prepared and aligned for processing by each of the layers but also help expand the layers' capacity to capture more features effectively.

**Discriminator ( $D$ ).** For our  $D$ , we employ an MLP architecture with fully connected Linear units. The primary role of  $D$  is to discern and evaluate temporal and spatial disparities between real and generated sensor data.

The architecture of  $D$  consists of multiple fully connected Linear layers. These layers are instrumental in expanding the input data, allowing it to be processed and transformed in a manner that facilitates the discrimination between real and fake sensor data. By employing Linear layers, we enable  $D$  to capture both local and global relationships in the input data.

Subsequently, a sigmoid activation function is applied to the output of  $D$ . It assigns a probability score for each data point within the window of sensor data. These The probability scores serve as an indicator of the likelihood that a given data point is real or fake.

In summary, our choice of an MLP architecture with fully connected Linear layers for  $D$  enables it to effectively assess the temporal and spatial differences between real and generated sensor data. The sigmoid activation function aids in quantifying the authenticity of individual data points within the window data, allowing  $D$  to play a crucial role in the adversarial training process.

## 4.3 Model Training

We train DSD-GAN in a two-player minimax game, where  $G$  aims to fool  $D$ , and  $D$  seeks to correctly identify real and fake data. During this process, we use the Adam optimizer to calculate corresponding weights for both models as  $\beta_D$  and  $\beta_G$ . Furthermore, the Smooth L1 Loss function is used for backpropagation to combine the benefits of both mean squared error and mean absolute error losses and minimize the influence of the outliers.

**4.3.1 Sensor Data Flow Extraction.** We use raw data consisting of readings obtained from IMU drone sensors to train proposed DSD-GAN. Since sensor data is in time-series form, we apply windowing to extract  $N$  vectors using a sliding approach. This process helps preserve temporal relations for individual sensors. Furthermore, we stack obtained time-series vectors on top of each other to capture a spectral relationship between sensor readings.

**4.3.2 GAN Training.** The generator and discriminator of our GAN are trained concurrently. For each epoch of the training, the generator produces a new set of samples which is then evaluated by the discriminator. The training data fed to the discriminator consists of the generator output samples (i.e., data generated from latent space random samples) and real sensor data. We design a custom loss function that is used to update the discriminator model's weight according to its performance. Subsequently, the generator weights are updated using a separate loss function based on the backpropagation of the discriminator feedback as indicated in Figure 4.

## 4.4 Ensemble Falsified Sensor Data

In this section, we expand our DSD-GAN model to an ensemble DSD-GAN (EDSD-GAN) for improved data generation (Figure 5).

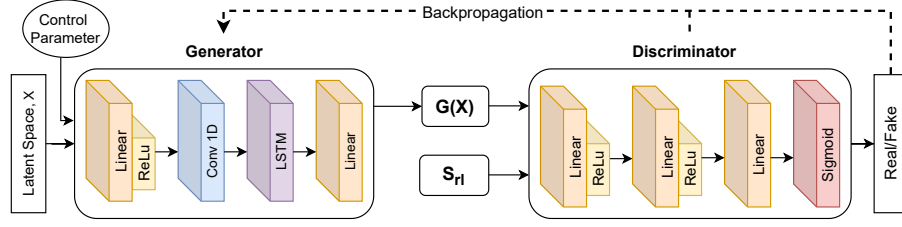


Figure 4: Model design of DSD-GAN.

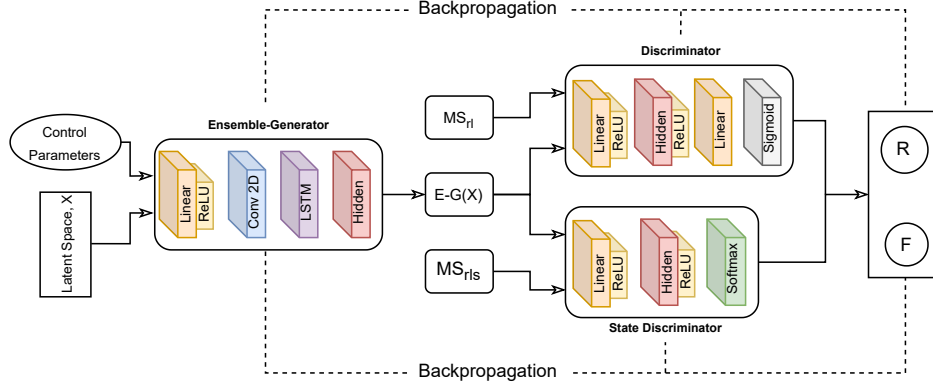


Figure 5: Model design of EDSD-GAN.

We introduce a state discriminator to enable the generator to learn intricate connections between sensor outputs over time, reflecting various flight phases such as take-off, standard flight, and landing. This approach enhances data authenticity, surpassing our previous method of merging separately generated sensor streams. It enables the model to develop a comprehensive understanding of sensor relationships, resulting in more realistic sensor data for each feature.

**4.4.1 State Discriminator Architecture.** The new addition of a state discriminator model is designed to predict the flight state (e.g. take-off, standard flight, and landing) from the sensor data. As input, the state discriminator takes in the multi-dimensional sensor data,  $MS_{rls}$ , consisting of both sensor data and state values. The architecture of the state discriminator is shown in Figure 5.

**4.4.2 Conditioning the Generator.** In EDSD-GAN, we modify the architecture of the generator by replacing 1D convolution with a 2D convolution layer, which results in a multi-dimensional array denoted as  $E-G(X)$ . Furthermore, to enable the generator to produce sensor data aligned with specific flight states (e.g. take-off, standard flight, and landing), we incorporate the state discriminator into the training process. The state label is concatenated with the random noise vector that serves as the generator's input.

**4.4.3 Training Procedure.** Our training procedure consists of the following steps:

**State Discriminator Training:** We train the state discriminator to accurately predict flight states from the sensor data. The state discriminator is provided with both real sensor data and generated data from the generator. The loss is computed using a smooth L1 loss function between the discriminator's predictions and the corresponding flight state labels. This step ensures that the state discriminator can effectively distinguish different flight states.

**Discriminator Training:** The Discriminator model is left mostly the same, however, it is altered to take in the real multi-dimensional sensor values,  $MS_{rl}$  and  $E-G(X)$ .

**Generator Training with State Conditioning:** We modify the generator training to include state conditioning. During each iteration, the generator takes a combination of random noise and the flight state label as input. We calculate the loss for the generator based on the output of the state discriminator when evaluating the generated data.

**Ensemble Falsified Sensor Data Generation:** After training, we generate ensemble falsified sensor data that captures different flight states. We employ the trained generator to produce sensor data samples corresponding to each flight state. Bypassing these generated samples through the state discriminator, we evaluate the extent to which the generator captures the desired flight states.

**Improved Training with State Feedback:** To further enhance the generator's performance, we introduce a feedback mechanism using the state discriminator's output. The state feedback loss is combined with the traditional adversarial loss during generator training. A hyperparameter, denoted as  $\lambda_{state}$ , controls the balance between these two loss components.

## 5 PERFORMANCE EVALUATION

In this section, we describe experimental settings and provide an extensive series of experiments to assess the performance of our system. We also make our code available at Github<sup>1</sup>.

### 5.1 Experimental Setting

**5.1.1 Dataset Description.** We utilize a publicly available drone sensor dataset by Whelan et al. [20] for training and evaluating

<sup>1</sup><https://github.com/mkuludag/DroneGAN-Synthetic-Sensor-Time-Series-Data-Generation-using-GANs>



our attack method against existing anomaly detectors. This dataset includes diverse sensor data from a Holybro S500 drone running PX4 Autopilot v1.11.3, comprising CSV logs from benign flights and flights with simulated GPS spoofing and Ping Denial of Service attacks. The attacks involve fabricated GPS messages and MAVLink PING message flooding, implemented using HackRF software-defined radio. Focusing on gyroscope and accelerometer IMU readings, our dataset consists of 20,000 consecutive benign readings and 20,000 equally distributed anomalous samples (jammed and spoofed) for training and testing anomaly detectors, with a 70/30 training-testing split.

**5.1.2 Anomaly Detectors.** Our proposed FDI attack is assessed against two drone sensor anomaly detectors, deployable on-board or off-board. First, we replicate the CNN-based classifier by Galvan et al. [6], using sequences of consecutive sensor readings to detect anomalous drone behavior. Additionally, we implement an autoencoder model, a choice found in several literature works [14, 19], consisting of encoder and decoder components with two Dense layers each (16 and 32 units). Trained to reconstruct benign samples, the model compares them with a predefined threshold for anomaly detection, relying on the mean squared logarithmic error (MSLE).

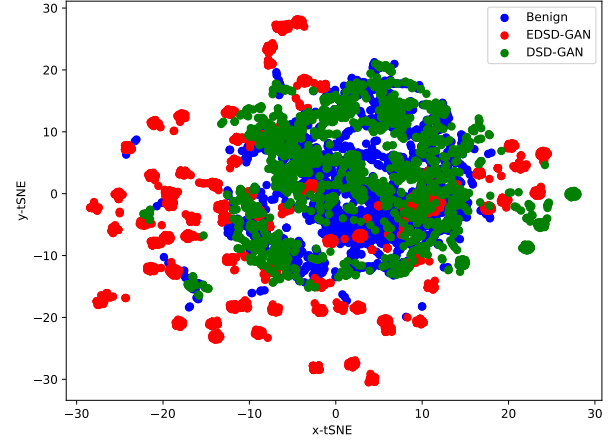
**5.1.3 Evaluation Metrics.** We use various metrics to assess anomaly classifier performance in the presence of the attack, with a focus on the false positive rate (FPR) and false negative rate (FNR). FPR quantifies false alarms, assessing the classifier's ability to classify benign samples while minimizing false alerts correctly. FNR is used to compute the attack success rate (ASR), indicating successful FDI attacks.

## 5.2 Data Quality Analysis

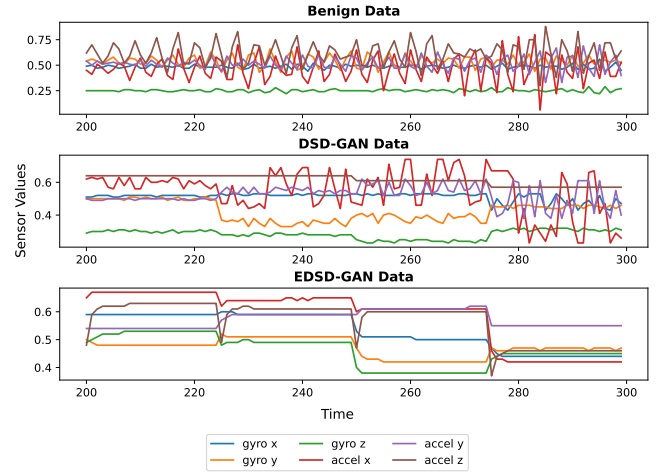
In this subsection, we assess the generated sensor data quality of DSD-GAN and EDSD-GAN. Training both models on benign data, we generate new data for gyroscope X, gyroscope Y, gyroscope Z, accelerometer X, accelerometer Y, and accelerometer Z sensors across 5000 consecutive timestamps. Employing t-SNE [18] for analysis, we visualize the distributions of generated and benign data in 2-D space.

Figure 6 compares the data generated by EDSD-GAN with benign and DSD-GAN data, illustrating the proximity of DSD-GAN data to benign data. This visualization sets the stage for our subsequent step, where we subject these synthetic datasets to real anomaly detectors to assess their deceptive potential. The graphical representation underscores the model's capability to create synthetic data resembling real sensor patterns across a comprehensive range of readings, anticipating the forthcoming analysis and testing against anomaly detectors.

We delve deeper into data distributions to identify the underlying reasons for the predominant clustering of EDSD-GAN data at the fringes of benign data distribution. As depicted in Figure 7, EDSD-GAN data exhibits notably less variation across all sensors over time. This suggests that the generator model yields similar results for diverse data inputs, resulting in reduced diversity. The visualization also reveals that while DSD-GAN captures more distinct distributions for each sensor, EDSD-GAN tends to smooth these distributions in relation to each other and across time.



**Figure 6: The t-SNE Visualization of Sensor Data: Comparative Analysis of Benign Data, SD-GAN Data, and EDSD-GAN Data for Gyroscope and Accelerometer sensor readings.**



**Figure 7: Visualization of Sensor Data over Time: Comparative Analysis of Benign Data, SD-GAN Data, and EDSD-GAN Data for Gyroscope and Accelerometer Sensor Readings**

## 5.3 Evaluation of Attack Effectiveness Against Anomaly Detectors

We conducted an extensive evaluation of the effectiveness of the GAN-based attack across multiple scenarios of False Data Injection (FDI) attacks targeting gyroscope and accelerometer sensors. Initially, we trained both CNN and autoencoder models to attain benchmark accuracies of 100% and 97.3%, respectively. Subsequently, we leveraged a single Deep Spatial Distribution Generative Adversarial Network (DSD-GAN) to generate synthetic data for six sensor readings independently, encompassing gyro X, gyro Y, gyro Z, accel X, accel Y, and accel Z. This synthetic data enabled us to evaluate classifier performance under minimal intrusion FDI attacks, where a single sensor is compromised. Following this, we devised scenarios where an attacker injected GAN-generated data for two or more sensor readings. Specifically, we identified four sensor combinations for both gyroscope and accelerometer data, along

with a scenario where all sensor readings were GAN-generated. We refer to these experiments as multi-sensor FDI attacks. Finally, we conducted a comparative analysis between the performance of individual DSD-GANs and an ensemble GAN in the context of multi-sensor scenarios.

**5.3.1 Single Sensor Attack.** Figure 8 represents the ASR of the DSD-GAN for minimal intrusion FDI attack. We observe that ASR has a similar tendency for corresponding sensors when tested against autoencoder and CNN anomaly detectors.

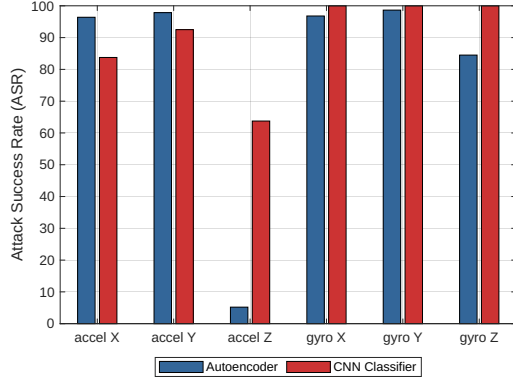


Figure 8: ASR for the single sensor FDI.

We observed that both anomaly detectors exhibited limitations in successfully identifying GAN-based False Data Injection (FDI) attacks targeting gyro X and gyro Y sensor values. Conversely, we noted that attacks involving the accel Z sensor were more likely to be detected, particularly in the case of the autoencoder. Furthermore, our proposed FDI attack demonstrated a high level of effectiveness when applied to accel X and accel Y sensors against the autoencoder, as well as for the gyro Z sensor when applied against the CNN classifier. In summary, we conclude that GAN-based single-sensor FDI attacks are more successful against the CNN classifier.

**5.3.2 Multi-Sensor Attack.** We implemented multi-sensor scenarios using two distinct approaches. Initially, each sensor value was generated independently using Deep Spatial Distribution Generative Adversarial Networks (DSD-GANs). Subsequently, we harnessed an ensemble of DSD-GANs to jointly generate various combinations of sensor values (e.g., accel (X,Y,Z), gyro (X,Y), and all sensors).

Figure 9 illustrates that in all cases, employing an ensemble of GANs for multi-sensor FDI led to higher ASR. Notably, manipulating accelerometer X and Z values jointly resulted in a lower ASR compared to the other accelerometer cases. For all FDI attacks conducted on the gyroscope sensor, we achieved a 100% ASR.

Similarly, an FDI attack executed against the autoencoder using an ensemble of DSD-GANs is more successful compared to separate GANs. From Figure 10, we identify that autoencoder is highly effective against FDI attacks that include fake accelerometer Z sensor values. Nonetheless, we emphasize that ensemble GANs allows to increase ASR for those cases at least twice.

## 5.4 Discussion

The results of our experiments demonstrate the remarkable effectiveness of adversarially generated data using GANs in deceiving real-time drone sensor anomaly detectors.

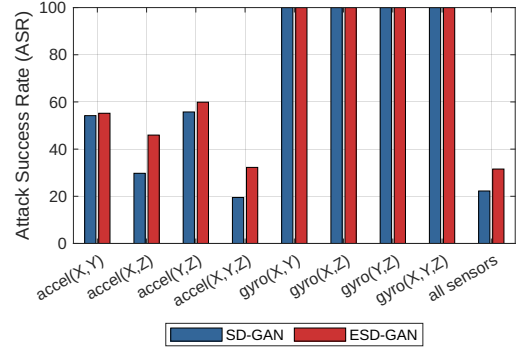


Figure 9: ASR for the multi-sensor FDI using separate DSD-GANs and EDSD-GANs when evaluated using CNN Classifier.

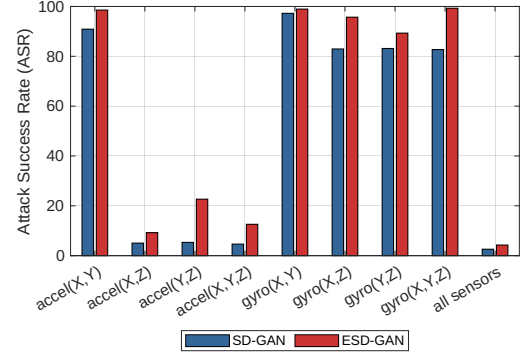


Figure 10: ASR for the multi-sensor FDI using separate DSD-GANs and EDSD-GANs when evaluated using Autoencoder.

**5.4.1 Single-sensor FDI.** In our analysis, we observed that both classifiers exhibit high sensitivity to FDI attacks targeting the accelerometer Z sensor. This heightened sensitivity can be attributed to the broader distribution of accelerometer Z sensor values, which enables the classifiers to better capture the benign data distribution compared to other sensors. Notably, our GAN-based FDI attack demonstrates consistent and effective performance when applied to other sensors, achieving an average ASR exceeding 90% against both anomaly detectors. This underscores the attack's robustness and its ability to succeed consistently in adversarial scenarios.

**5.4.2 Multi-sensor FDI.** In our investigation, we assessed two key aspects of multi-sensor FDI scenarios: 1) the impact of concurrently injecting synthetic data generated by GANs into multiple sensors and 2) the advantages of employing an ensemble of GANs for multi-sensor data injection.

As the number of sensors injected with GAN-generated data increases, FDI attack ASR decreases. Findings reveal that injecting fabricated data for specific sensor attributes, such as accelerometer Z, significantly reduces ASR, indicating the influential role of certain sensor attributes in machine learning classifiers' decision-making. Enhanced classifier performance against FDI attacks using multiple sensors is attributed to their ability to detect inconsistencies in correlations among injected GAN-generated sensor values.

Utilizing an ensemble of DSD-GANs to capture spatial relationships between sensor values correlates with increased ASR for all multi-sensor FDI attacks. The ensemble's impact is notable, particularly when ASR is initially lower for separate DSD-GANs with

sensor combinations like accel (X,Y), accel (X,Y,Z), and all sensors. Despite the lower initial ASR, employing an ensemble leads to a minimum doubling of ASR. In the case of an FDI attack targeting accel (Y, Z) sensors against the autoencoder classifier, the ensemble of GANs elevates ASR from 5.35% to 22.68%.

**5.4.3 Analysis of Anomaly Detectors.** Our main goal was to develop an effective FDI attack using an ensemble of GANs. Initially, the CNN classifier outperformed the autoencoder on a benchmark dataset with benign, spoofed, and jammed sensor readings. However, further experiments revealed the autoencoder's greater effectiveness in detecting GAN-generated injected data for the majority of multi-sensor FDI scenarios.

Hence, we conclude that the CNN classifier demonstrates less effectiveness in identifying anomalies it hasn't encountered before, necessitating frequent retraining to capture novel types of FDI attacks. This observation aligns with the structural differences between the two classifiers: the CNN classifier is trained on both benign and anomalous samples, while the autoencoder exclusively learns patterns from benign data. Therefore, while the autoencoder anomaly detector may not prevent all FDI attacks, it is more robust to a variety of anomaly types.

Both classifiers exhibited high sensitivity to deviations in accelerometer Z sensor values, suggesting the reliance of anomaly detectors on these values. Based on this insight, we suggest that GANs can be used to enhance model explainability via (1) incorporating generated data during training and (2) reducing dependence on the single feature and aiding in feature selection for anomaly detection.

Conversely, attackers can exploit gathered intelligence about classifier performance to launch a poisoning attack. Injecting false data, especially for significant features like accel Z, can render the model ineffective and result in model drift. We also demonstrated that avoiding the injection of the most significant features (i.e., accel Z) results in higher ASR, as indicated in Figures 9 and 10.

## 6 CONCLUSION

In our investigation into UAV security, we unveiled the vulnerabilities within Machine Learning-based anomaly detection systems. We introduced two potent GAN-based FDI attacks, targeting critical IMU sensors of drones. The DSD-GAN model skillfully manipulated individual drone sensor data by exploiting temporal features, resulting in compelling FDI attacks. Additionally, our innovative EDSD-GAN, utilizing an ensemble approach to preserve spatial sensor relationships, excelled in crafting high-fidelity multi-sensor FDI attacks. In our evaluations against CNN and autoencoder-based anomaly detectors, we discovered these classifiers exhibited remarkable susceptibility to DSD-GAN FDI attacks on single sensors, failing to detect anomalies in nearly 80% of cases, on average. Complementing this, the EDSD-GAN demonstrated its deceivability, achieving over a 50% increase in the attack success rate compared to separate DSD-GANs. Our work also identified that anomaly detectors trained on exclusively benign data are more robust against unseen FDI attacks. Finally, we discussed the potential alternatives for utilizing developed approaches as a tool for anomaly detector analysis.

## REFERENCES

- [1] Alankrita Aggarwal, Mamta Mittal, and Gopi Battineni. 2021. Generative adversarial network: An overview of theory and applications. *International Journal of Information Management Data Insights* 1, 1 (2021), 100004.
- [2] Sandra Pérez Arteaga, Luis Alberto Martínez Hernández, Gabriel Sánchez Pérez, Ana Lucila Sandoval Orozco, and Luis Javier García Villalba. 2019. Analysis of the GPS Spoofing Vulnerability in the Drone 3DR Solo. *IEEE Access* 7 (2019), 51782–51789. <https://doi.org/10.1109/ACCESS.2019.2911526>
- [3] Jabang Aru Saputro, Esa Egistian Hartadi, and Mohamad Syahrul. 2020. Implementation of GPS Attacks on DJI Phantom 3 Standard Drone as a Security Vulnerability Test. In *2020 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering (ICITAMEE)*, 95–100. <https://doi.org/10.1109/ICITAMEE50454.2020.9398386>
- [4] Wenxin Chen, Yingfei Dong, and Zhenhai Duan. 2019. Manipulating Drone Position Control. In *2019 IEEE Conference on Communications and Network Security (CNS)*, 1–9. <https://doi.org/10.1109/CNS.2019.8802817>
- [5] Wenxin Chen, Zhenhai Duan, and Yingfei Dong. 2017. False data injection on EKF-based navigation control. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, 1608–1617. <https://doi.org/10.1109/ICUAS.2017.7991406>
- [6] Julio Galvan, Ashok Raja, Yanyan Li, and Jiawei Yuan. 2021. Sensor Data-Driven UAV Anomaly Detection using Deep Learning Approach. In *MILCOM 2021 - 2021 IEEE Military Communications Conference (MILCOM)*, 589–594. <https://doi.org/10.1109/MILCOM52596.2021.9653036>
- [7] Jiaxin Gao, Qian Zhang, and Jiyang Chen. 2020. EKF-based actuator fault detection and diagnosis method for tilt-rotor unmanned aerial vehicles. *Mathematical Problems in Engineering* 2020 (2020).
- [8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.
- [9] Dingfei Guo, Maiying Zhong, and Donghua Zhou. 2018. Multisensor Data-Fusion-Based Approach to Airspeed Measurement Fault Detection for Unmanned Aerial Vehicles. *IEEE Transactions on Instrumentation and Measurement* 67, 2 (2018), 317–327. <https://doi.org/10.1109/TIM.2017.2735663>
- [10] Samurdi Karunaratne, Enes Krijestorac, and Danijela Cabric. 2021. Penetrating RF fingerprinting-based authentication with a generative adversarial attack. In *ICC 2021-IEEE International Conference on Communications*, IEEE, 1–6.
- [11] Kyo Kim, Siddhartha Nalluri, Ashish Kashinath, Yu Wang, Sabin Mohan, Miroslav Pajic, and Bo Li. 2020. Security analysis against spoofing attacks for distributed UAVs. *Decentralized IoT Systems and Security* (2020).
- [12] Heng Li, ShiYao Zhou, Wei Yuan, Jiahuan Li, and Henry Leung. 2019. Adversarial-example attacks toward android malware detection system. *IEEE Systems Journal* 14, 1 (2019), 653–656.
- [13] Savvas Papaioannou, Panayiotis Kolios, Christos G. Panayiotou, and Marios M. Polycarpou. 2020. Cooperative Simultaneous Tracking and Jamming for Disabling a Rogue Drone. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 7919–7926. <https://doi.org/10.1109/IROS45743.2020.9340835>
- [14] Seunghyoung Ryu, Jiyeon Yim, Junghoon Seo, Yonggyun Yu, and Hogeon Seo. 2022. Quantile Autoencoder With Abnormality Accumulation for Anomaly Detection of Multivariate Sensor Data. *IEEE Access* 10 (2022), 70428–70439. <https://doi.org/10.1109/ACCESS.2022.3187426>
- [15] Lwin Khin Shar, Wei Minn, Nguyen Binh Duong Ta, Jiani Fan, Lingxiao Jiang, and Daniel Lim Wai Kiat. 2022. DronLomaly: runtime detection of anomalous drone behaviors via log analysis and deep learning. In *2022 29th Asia-Pacific Software Engineering Conference (APSEC)*, IEEE, 119–128.
- [16] Yi Shi, Kemal Davaslioglu, and Yalin E. Sagduyu. 2019. Generative Adversarial Network for Wireless Signal Spoofing. In *Proceedings of the ACM Workshop on Wireless Security and Machine Learning* (Miami, FL, USA) (WiseML 2019). Association for Computing Machinery, New York, NY, USA, 55–60. <https://doi.org/10.1145/3324921.3329695>
- [17] Muhammad Usama, Muhammad Asim, Siddique Latif, Junaid Qadir, and Ala-Al-Fuqaha. 2019. Generative Adversarial Networks For Launching and Thwarting Adversarial Attacks on Network Intrusion Detection Systems. In *2019 15th International Wireless Communications Mobile Computing Conference (IWCMC)*, 78–83. <https://doi.org/10.1109/IWCMC.2019.8766353>
- [18] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).
- [19] Jason Whelan, Thanigajan Sangarapillai, Omar Minawi, Abdulaziz Almelhamdi, and Khalil El-Khatib. 2020. Novelty-Based Intrusion Detection of Sensor Attacks on Unmanned Aerial Vehicles. In *Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks* (Alicante, Spain) (Q2SWinet '20). Association for Computing Machinery, New York, NY, USA, 23–28. <https://doi.org/10.1145/3416013.3426446>
- [20] Jason Whelan, Thanigajan Sangarapillai, Omar Minawi, Abdulaziz Almelhamdi, and Khalil El-Khatib. 2020. UAV Attack Dataset. IEEE Dataport. <https://doi.org/10.21227/00dg-0d12>