Novelty-based Intrusion Detection of Sensor Attacks on Unmanned Aerial Vehicles

Jason Whelan
jason.whelan@ontariotechu.net
Ontario Tech University
Ontario, Canada

Thanigajan Sangarapillai
thanigajan.sangarapillai@ontariotechu.net
Ontario Tech University
Ontario, Canada

Omar Minawi
omar.minawi@ontariotechu.net
Ontario Tech University
Ontario, Canada

Abdulaziz Almehmadi
aalmehmadi@ut.edu.sa
University of Tabuk
Tabuk, Saudi Arabia

Khalil El-Khatib
khalil.el-khatib@uoit.ca
Ontario Tech University
Ontario, Canada

ABSTRACT

Unmanned Aerial Vehicles (UAVs) have proven to be a useful technology in numerous industries including industrial control systems surveillance, law enforcement, and military operations. Due to their heavy reliance on wireless protocols and hostile operating environments, UAVs face a large threat landscape. As attacks against UAVs increase, an intelligent intrusion detection system (IDS) is needed to aid the UAV in identifying attacks. The UAV domain presents unique challenges for intelligent IDS development, such as the variety of sensors, communication protocols, UAV platforms, control configurations, and dataset availability. In this paper, we propose a novelty-based approach to intrusion detection in UAVs by using one-class classifiers. One-class classifiers require only non-anomalous data to exist in the training set. This allows for the use of flight logs as training data, which are created by most UAVs during flight by default. Principal Component Analysis is applied to sensor logs for dimensionality reduction, and one-class classifier models are generated per sensor. A number of one-class classifiers are selected: One-Class Support Vector Machine, Autoencoder Neural Network, and Local Outlier Factor. The pre-processing, feature selection, training, and tuning of the selected algorithms is discussed. GPS spoofing is used throughout the paper as a common example of an external sensor-based attack. This approach shows to be effective across multiple UAV platforms with platform-specific F1 scores up to 99.56% and 99.73% for benign and malicious sensor readings respectively.

CCS CONCEPTS

• **Security and privacy** → **Intrusion detection systems**; • **Computer systems organization** → *Sensors and actuators*; • **Computing methodologies** → *Machine learning*; • **Theory of computation** → Semi-supervised learning.

KEYWORDS

intrusion detection; novelty detection; machine learning; unmanned aerial vehicles; robotic vehicles; cyber-physical systems

ACM Reference Format:
Jason Whelan, Thanigajan Sangarapillai, Omar Minawi, Abdulaziz Almehmadi, and Khalil El-Khatib. 2020. Novelty-based Intrusion Detection of Sensor Attacks on Unmanned Aerial Vehicles. In *16th ACM Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet'20), November 16–20, 2020, Alicante, Spain.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3416013.3426446

Q2SWinet'20, November 16–20, 2020, Alicante, Spain

ACM ISBN 978-1-4503-8120-8/20/11…$15.00
https://doi.org/10.1145/3416013.3426446

1 INTRODUCTION

As technology advances, many tasks once performed by humans are now being fulfilled by computers. The aircraft industry is not exempt from this, as autonomous Unmanned Aerial Vehicles (UAVs) replace tasks that were once performed by humans. The development and application of UAVs are rapidly expanding across a variety of high-risk sectors including industrial control systems surveillance, law enforcement, and military operations. Like other autonomous systems, this is due to their ability to perform with a high degree of precision, with minimal human intervention. The inherent risk in industries such as emergency management or military operations allows UAVs to be an effective human replacement, as not only do they negate the risk to the lives of the operators, they are less susceptible to human error.

Although used in high-risk environments, UAVs are prone to cyber security attacks [17]. They primarily operate in open and potentially hostile areas, and utilize wireless communication technologies to transmit and receive data from a Ground Control Station (GCS), or from on-board sensors. Considering that the flight controller uses this data as input, the potential damage that could be caused by attacks drastically increases.

As attacks on UAVs increase in frequency, there is a strong need for protective measures against attacks on such systems. Intrusion Detection Systems (IDS) are a potential solution to the problem at hand. Since the threat landscape for UAVs is ever changing as technology progresses, traditional signature-based IDS detection will not be able to effectively protect the UAV against threats. Depending on the mission, different UAV platforms can be used. This may introduce differences in many parameters, including payloads, required sensors, control configurations, to list a few. As such, this paper proposes an IDS that utilizes novelty detection methods for

detecting security threats. Novelty detection, rather than anomaly detection, provides the ability to learn from a training dataset where anomalies are not present. This allows for the utilization of existing flight logs to learn what is normal for the specific UAV the IDS will operate on.

This paper makes the following contributions:

- Proposes the use of one-class novelty detection techniques for intrusion detection in unmanned aerial vehicles.
- Discusses how one-class classifiers can be used to solve the unavailability of labelled UAV intrusion detection datasets by learning from existing flight logs.
- Demonstrates the performance of various classifiers using data from simulated flights of various UAV platforms.

The rest of the paper is organized as follows: Section 2 provides a background on the research problem and related work. Section 3 discusses the one-class novelty detection approach in depth. The performance evaluation of the selected methods, including the experiment design, simulation of attacks, and collection of associated flight logs are discussed in Section 4. Finally, we draw our conclusion and discuss future work in Section 5.

2 BACKGROUND

Intrusion detection techniques can be classified as signature-based, specification-based, statistical-based, anomaly-based or hybrids [4]. Signature-based IDS use a predefined set of "signatures" that are created from patterns in malicious traffic or behaviours. These types of IDS rely completely on detecting known attacks for which signatures have been created. In the case of a UAV, it would be an extremely difficult and arduous task to create and maintain a signature set that covers the vast majority of UAVs and attacks. Attacks would need to be conducted on each type of UAV, recorded, and signatures engineered. It is likely for these reasons that we cannot find any examples of a UAV IDS solely based on signatures.

Mitchell and Chen [10] have proposed a specification-based UAV IDS called Behavior Rule-based UAV Intrusion Detection System (BRUIDS). Specification-based intrusion detection relies upon predefined behaviour-based rules. For example, a set of rules may be created to define the path a UAV must take in order to detect hijacking attacks. When an attack such a GPS spoofing causes the UAV to go off course, the specification-based IDS would be triggered. These types of IDS work well in scenarios where reactions to an attack can easily be defined and the system they are implemented on is standardized. In the UAV domain, many different components are used, mission paths can change with each flight, and operating bounds vary from platform to platform. Devices can have varying normal bounds, due to factors such as the version of flight controller software, the number of active components, or environmental factors. These limitations make specification-based IDS difficult to maintain in the UAV domain.

Bayesian and game-theoretic approaches are popular statistics-based approaches applied to UAV IDS [11, 12, 14]. These approaches create Bayesian games where attackers and defenders are the participants. Used in UAV networks, the IDS allows malicious actors to be identified and ejected. These solutions work well as low power techniques, however, they are only applicable to UAV networks as they rely on information from other game participants in the UAV network. Wang and Feng [16] uses a game theory approach by "protecting" only sensors that are likely to be attacked, with the aim of reducing power consumption. The Nash equilibrium solutions of the non-cooperative games provide the UAV with insight into what sensor is likely at risk of an attack. Unfortunately, the consequence of an incorrect prediction can leave the UAV vulnerable.

Anomaly-based techniques allow for the accurate detection of known and unknown intrusions. Anomaly detection is the process of identifying data that is outside the expected normal behaviour. In the case of an IDS, anomaly detection aims to identify malicious communications and actions that differ from what is expected to be normal UAV operations. Detecting anomalies is a broad field in itself, and is used in various other areas such as fraud detection, industrial damage detection, system health monitoring, and more [3]. There are many techniques for detecting anomalies, each with their own strengths and limitations. These techniques are not overly popular in the UAV IDS domain, likely because of the lack of universal training data. The use of a long short-term memory (LSTM) autoencoder and vanilla autoencoder for this task have been recently introduced by Bae and Joe [1]. Unfortunately, this paper lacks significant detail such as flight data acquisition method, feature selection, pre-processing, and performance evaluation against attacks.

Hybrid approaches combine more than one detection technique with the intent of increasing efficiency while also retaining the ability to detect unknown attacks. To remain efficient, however, they require ongoing research of attacks and maintenance for signature creation [5, 13].

3 ONE-CLASS NOVELTY DETECTION

Previous works propose UAV IDS solutions that can be effective on specific platforms or in specific environments and scenarios. However, the UAV domain presents a number of challenges for IDS development such as the wide variety of sensors, UAV platforms, communication protocols, control configurations, and more. This has lead to a lack of universal training data for machine learning-based approaches, which are necessary to protect the UAV from known and unknown attacks with accuracy. In addition, other approaches incur implementation overhead and maintenance, making them time and cost prohibitive.

Anomaly-based approaches can be effective solutions to the aforementioned challenges [9]. Unfortunately, they require anomalies to pre-exist in the dataset when training. This then reintroduces the problem of dataset availability and the diversity of technologies used. By implementing a novelty-based approach, existing flight logs can be utilized by a classifier to learn an underlying distribution of sensor values. New observations outside of this distribution are classified as novelties, and therefore potentially malicious. The proposed approach requires only flight logs from a "training flight" in which the UAV conducts a mission where no attacks occur.

3.1 Evaluated Methods

We choose classifiers that provide the ability to perform one-class novelty detection. One-class novelty detection is not reliant on having data on all current attack vectors. Rather, this approach

focuses on learning normal behaviour and classifying messages that appear outside the normal. This provides a comprehensive solution, which allows our approach to detect known and unknown attacks. In this paper, we discuss the use of One-Class Support Vector Machines, Autoencoder Neural Network, and Local Outlier Factor.

Support Vector Machines (SVMs) are supervised algorithms that can be used for classification. An SVM algorithm is given two sets of data, and tries to create a model to separate categories through a gap as wide as possible. The SVM then makes decisions based on which side of the gap a certain point is placed. In a One-Class SVM (OC-SVM), the model is only trained on normal data, absent of malicious activities. Therefore, any data that is deemed as an anomaly is classified as an intrusion.

An Autoencoder Neural Network functions by taking input, compressing it and trying to reproduce the same input. The Autoencoder is comprised of the following four parts: input data, encoding function, decoding function and loss function. The input data is the data that is getting encoded and decoded. The encoding function takes the input data, and encodes it. The decoding function takes the encoded input and decodes it. The loss function is responsible for evaluating how optimal the autoencoder is performing. The autoencoder functions at a high level of success when data is encoded then decoded and the result is very close to the original data.

Local Outlier Factor is an unsupervised algorithm which can be used for anomaly detection. This algorithm computes the local density deviation of a given point, in relation to n number of neighbors. Outliers are found to have significantly less density than their neighbors.

3.2 Pre-Processing

3.2.1 Flight Log Feature Extraction and Clustering. Flight logs are an ideal source of data for a machine learning-based IDS. They are readily available by default, contain sensor readings throughout the flight, and are written to in real time for inference. Most if not all implementations of UAVs will keep a flight log with information regarding the UAV's various sensors and components. Since this is common among various models and implementations of UAVs, using flight logs as input can make a machine learning-based IDS viable across the industry. Rather than feeding all the data in the log into a singular model, there is a need to reduce dimensionality as much as possible. The significant number of features and data in logs can slow down the training and inference of a model. A large portion of the data may not be useful, so there is a need to eliminate such data. This is performed by clustering features that are unique to different types of sensors/components, and training separate models for each cluster. Any unrelated or autopilot-specific features are dropped. With a separate machine learning model for each sensor and component, there is the added advantage of being able to determine which sensor or component is being targeted. Unlike other approaches, this allows for the detection not only of the occurrence of an attack, but the potential to classify the type of attack or sensor being targeted.

In this specific implementation, the flight log can be split into multiple CSVs based on the sensor/topic logged. Certain topics may not be polled at the same rate as others, so there is also the need to interpolate certain values.

For the sake of brevity, features were only clustered for the GPS component. Some key features that belong to this cluster are *latitude*, *longitude*, *altitude*, as well as velocity and position data. These features are common among all UAVs, so the IDS is not limited to specific models. The feature extraction and clustering follows Algorithm 1. For the purpose of this paper, an additional step was added to label the data.

Algorithm 1: Feature Extraction and Clustering

```
Result: CSV dataset per cluster
initialization;
for CSV from flight log do
    if CSV is related to sensor then
        Merge CSV into sensor's dataframe with
          key=timestamp;
    end
    ;
end
set index to timestamp and sort;
Interpolate NaN values using previous or next values;
Remove any columns with infinity or NaN values;
```

3.2.2 Principal Component Analysis. After performing clustering and feature extraction, a cluster can still contain 50-150 features. To reduce the dimensionality of the dataset and to improve the performance and results of the machine learning algorithms, principal component analysis (PCA) was used. PCA is able to transform a set of features into principal components to better explain the variance in the original set. These principal components are created from linear combinations of the original features in order to capture a certain percentage of variance from the original set while reducing dimensionality. Useful data contains variance, and the more variance the data has, the higher it's importance. As it is able to reduce dimensionality to a few principal components, it can be used to visualize data. Figure 1 shows 3D scatter plot created by reducing a set of 85 features to only three principal components. The number of principal components can be specified by explicitly declaring a number, or by specifying how much variance to retain. The PCA for SVM and LOF are configured to capture 85% of variance from the dataset, while the autoencoder performed better with 95% variance. The training set is used to fit PCA. The fitted PCA is used to transform both the testing and training set. Both sets of data are normalized before being used to fit or be transformed by PCA.

3.3 Training and Tuning

Two datasets are used to train and test algorithms. A benign dataset, not containing malicious data, is used to train and fit the models. A separate dataset containing both benign and malicious data is used, for the purpose of this paper, to test and show performance results of the models.

3.3.1 One-Class Support Vector Machine. One-Class Support Vector Machine (OC-SVM) is a semi-supervised method where the

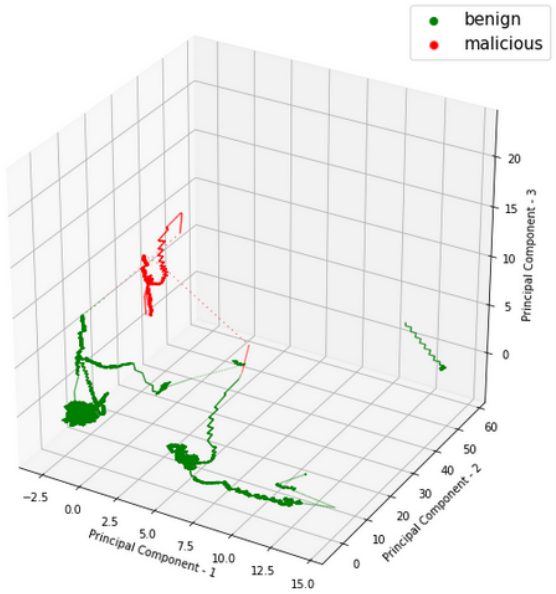

Figure 1: 3DR IRIS+ 3 PC Autoencoder Detection

classifier is trained only on the benign data, and classifies new data as being similar or different from the training set. A binary classification function is computed to determine where the probability density lies. Points outside of this support are classified as being outside of the class, and therefore are novelties.

OC-SVM requires two hyperparameters to be tuned in order to optimize performance, *nu* (ν) and *gamma* (γ). The hyperparameters used for each UAV platform are listed in Table 3. The Gaussian Radial Basis Function (RBF) kernel is used, due to its performance in one-class classification problems [2].

3.3.2 Autoencoder Neural Network. The autoencoder is used by training on a benign dataset using the mean squared error (MSE) loss function. The test dataset containing both benign and malicious data is then fed into the autoencoder. If the MSE of the reconstructed output is over a delta threshold, T, it is classified as a novelty. The thresholds used for each UAV platform are listed in Table 3.

An autoencoder neural network consisting of 25 input nodes, 3 hidden nodes, and 25 output nodes was used for testing. The model was trained for a maximum of 100 epochs using the Adam optimizer, with early stopping. The rectified linear activation function is used as it can output true zero values. This is important for the autoencoder as it allows a sparsity when learning the compressed representation [7].

3.3.3 Local Outlier Factor. Density-based methods such as the Local Outlier Factor (LOF) algorithm can be effective on lower dimension data. After PCA was applied, the dimentionality remained fairly high, however, we continued to use LOF as a frame of reference for performance evaluation. The LOF algorithm calculates the local density deviation from an input to its local neighbours. LOF requires the hyperparameter k, the number of neighbors, to be defined by the user as *k-neighbors*. Using this value, the distance to the kth point is defined as the *k-distance*. Next, the *reachability distance* is calculated as the maximum of the distance between two points and the *k-distance* of the given point. By using the *reachability distance*, the *local reachability density* is calculated to determine how far a point is from the next cluster of points. This value is compared to the value of its *k-neighbors* to get a sense of how dense a point is compared to the average. A lower density signifies an outlier. The specific *k-neighbors* values used for each UAV platform are listed in Table 3.

4 PERFORMANCE EVALUATION

4.1 Experiment Design

In order to conduct novelty detection experiments, multiple datasets are required. As one-class classifiers are trained on data where no anomalies are present, a "normal" flight log is needed. In addition, logs of flights where the UAV experiences an attack are also needed. All flights are autonomous survey missions, with a flight time between 10-30 minutes depending on the UAV platforms maximum velocity. UAVs of various platforms, control configurations, and sensors are used. The list of UAVs used is shown in Table 1.

Table 1: UAVs Used for Experiments

Platform	Model	Simulation Type
Quadcopter	3DR IRIS+	Software-in-the-loop
Quadcopter	Holybro S500	Hardware-in-the-loop
Hexacopter	Yuneec H480	Software-in-the-loop
VTOL	DeltaQuad VTOL	Software-in-the-loop
Tailsitter	Standard Tailsitter	Software-in-the-loop
Plane	Standard Plane	Software-in-the-loop

4.1.1 Attack Simulation. Simulations are conducted using the PX4 autopilot and Gazebo robotics simulator. PX4 is an open source autopilot firmware popular in the commercial space as the basis for many custom UAVs. The proposed approach makes use of logged sensor values that are ubiquitous across autopilots; any autopilot specific attributes or features are discarded. Various UAV platforms are used to ensure the interoperability of machine learning models across different UAVs. Hardware-in-the-loop and software-in-the-loop simulations are also conducted to ensure accuracy. The simulation environment consists of the autopilot firmware connected to the simulator. The simulator is responsible to simulating the environment, as it sends environmental sensor inputs to the autopilot, and reacts to the received actuator outputs.

Fail-safes are disabled during simulations as to not cause the UAV to alter course because of a lost connection or other unknown fault. This would skew the data by creating more anomalies than those from the attack itself.

For the sake of brevity, a single attack is shown as an example. GPS spoofing is a common sensor-based attack against UAVs, making it an obvious choice. The GPS system uses a network of satellites equipped with transmitters and synchronized clocks. Each satellite transmits the current system time, as well its orbital information. A receiver on Earth can utilize messages from four of the satellites in order to triangulate its location. Three of the messages are used to figure out its location. Since the speed of light can be assumed as a

constant, a GPS receiver can calculate the distance between itself and a satellite by comparing the message's time of arrival (TOA) to the time specified in the message. Three of these messages are enough to obtain the receiver's location, but a fourth message can further increase accuracy by providing height as well as an accurate time.

In a real world application, a malicious party can broadcast fake GPS signals, either by replaying a modified set of signals or by creating their own. This can cause the target GPS receiver to falsely estimate its position. Commonly, this occurs by broadcasting signals synchronized with genuine signals, then increasing the power of the fake signals, causing the receiver to switch over to receiving messages from the fake broadcaster. To simulate GPS spoofing accurately, an integration into the simulated environment is created. With this integration, we can broadcast spoofed GPS signals into the environment, which in turn are picked up and sent to the autopilots GPS sensor. Typically, the GPS sensor data in our environment is received by the autopilot at a rate of 5Hz. When the spoofed signals are injected into the simulation environment, they are sent at a much higher rate. This causes the autopilot to lock to the "stronger" signal [15]. After a lock is made, the UAVs position estimation will change, causing it to drift off course [8]. When the attack stops, the UAV locks back to the legitimate signal and moves to correct its position and resumes its trajectory on the mission. The UAV is subjected to the GPS spoofing attack for 30 seconds, enough to cause it to drift significantly.

4.1.2 Dataset Creation. Flight logs can contain various information and have various formats, however, common attributes are recorded for post-flight analysis. For our particular experiments, the autopilot used saves flight logs into ULOG format. This is a binary format consisting of header, definition, and data sections. The header contains the file magic number, log version, and timestamp. The definition sections contains the logged attributes and values themselves. The data section contains informational, debug, warning, and emergency information sent from the autopilot to the GCS. For our purposes, we extract the parameters from the definition section of the log by type resulting in multiple CSVs.

The logs are downloaded from the UAV after each flight, and timestamps are removed from all logs. Attack start and end times are recorded in order to assist in the labelling process. As we are using a one-class approach, labelling is only necessary for performance analysis. The number of labelled benign and malicious sensor values is shown in Table 2.

A deeper description of the simulation environment, attack simulation, and the the full dataset is available on IEEE DataPort [6].

Table 2: Dataset Description

Dataset Model	Benign	Malicious
3DR IRIS+	305140	6596
Holybro S500	349722	7164
Yuneec H480	54377	1123
DeltaQuad VTOL	18308	1111
Standard Tailsitter	17921	1113
Standard Plane	23198	1055

4.2 Results

When creating an intrusion detection system, we look to minimize the false positive rate while maximizing the detection rate. To analyze the performance of the proposed method, a number of metrics are computed for each algorithm. These metrics are the precision, recall, and F1 score. Precision represents the number of positive predictions that truly belong to the positive class. Recall represents the number of positive predictions out of every positive sample. The F1 score is indicative of overall performance, as the weighted average of both precision and recall. Each one of these metrics for each algorithm and its corresponding UAV platform is shown in Table 3.

The average F1 score of all benign and malicious classifications were computed to determine the overall highest performer. These results show the autoencoder as the the algorithm of choice for the proposed approach, with an average F1 score of 94.81%. Next, the OC-SVM performed with an average F1 of 81.17%. Last, LOF with an F1 of 58.93%.

5 CONCLUSION

UAVs perform critical tasks in high risk environments. Given the large threat landscape they face, an intelligent IDS is necessary to help detect cyber attacks. Unfortunately, due to the wide range of components and technologies used in UAV deployments, it becomes very difficult to create an accurate IDS that can be used with many different implementations. This paper has proposed the use of one-class classifiers to train a novelty-based IDS, which learns normal sensor values from previous flight logs. The experiments conducted in this paper show promising results, particularly with an autoencoder neural network, to detect sensor-based attacks against UAVs. The proposed approach also remains effective across a variety of UAV platforms and control configurations.

Our future work will focus on developing classifiers for a variety of sensors and to use the proposed approach in the development of an on-board IDS for UAVs. Placing IDS processing directly on-board the UAV has the potential to mitigate denial of service and jamming attacks even when the connection to the GCS is lost. This opens the door to not only the detection of attacks under duress, but also the potential for autonomous mitigation.

REFERENCES

[1] Gimin Bae and Inwhee Joe. 2020. UAV Anomaly Detection with Distributed Artificial Intelligence Based on LSTM-AE and AE. In *13th International Conference on Multimedia and Ubiquitous Engineering, MUE 2019 and 14th International Conference on Future Information Technology, Future Tech 2019*. Springer Verlag, 305–310.

[2] A. Bounsiar and M. G. Madden. 2014. Kernels for One-Class Support Vector Machines. In *2014 International Conference on Information Science Applications (ICISA)*. 1–4.

[3] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly detection: A survey. *ACM computing surveys (CSUR)* 41, 3 (2009), 15.

[4] Gaurav Choudhary, Vishal Sharma, Ilsun You, Kangbin Yim, Ray Chen, and Jin-Hee Cho. 2018. Intrusion Detection Systems for Networked Unmanned Aerial Vehicles: A Survey. In *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*. IEEE, 560–565.

[5] Jean-Philippe Condomines, Ruohao Zhang, and Nicolas Larrieu. 2019. Network intrusion detection system for UAV ad-hoc communication: From methodology design to real test validation. *Ad Hoc Networks* 90 (2019), 101759.

[6] Jason Whelan; Thanigajan Sangarapillai; Omar Minawi; Abdulaziz Almehmadi; Khalil El-Khatib. 2020. UAV Attack Dataset. https://doi.org/10.21227/00dg-0d12

Table 3: Performance Evaluation

Model	Classifier	Label	Precision	Recall	F1 Score	Hyperparameters
3DR IRIS+	OC-SVM	Malicious	0.62074	1.00000	0.76600	$\nu = 0.011$
		Benign	1.00000	0.98707	0.99335	$\gamma = 0.000211$
	LOF	Malicious	0.04801	1.00000	0.09162	$k\_neighbors = 1000$
		Benign	1.00000	0.57136	0.72722	
	Autoencoder	Malicious	0.75495	0.99909	0.86003	$T = 97.2\%$
		Benign	0.99998	0.99299	0.99647	
Holybro S500	OC-SVM	Malicious	0.69628	0.96482	0.80885	$\nu = 0.011$
		Benign	0.99927	0.99138	0.99531	$\gamma = 0.000211$
	LOF	Malicious	0.04571	1.00000	0.08742	$k\_neighbors = 3500$
		Benign	1.00000	0.57234	0.72801	
	Autoencoder	Malicious	0.64561	0.99707	0.78374	$T = 96.89\%$
		Benign	0.99994	0.98879	0.99433	
Yuneec H480	OC-SVM	Malicious	0.51196	0.99110	0.67516	$\nu = 0.0211$
		Benign	0.99981	0.98049	0.99006	$\gamma = 0.0003$
	LOF	Malicious	0.09658	1.00000	0.17614	$k\_neighbors = 3100$
		Benign	1.00000	0.80681	0.89308	
	Autoencoder	Malicious	0.83483	0.99020	0.90591	$T = 97.6\%$
		Benign	0.99980	0.99595	0.99787	
DeltaQuad VTOL	OC-SVM	Malicious	0.41419	0.99280	0.58453	$\nu = 0.1$
		Benign	0.99952	0.91479	0.95528	$\gamma = 0.000211$
	LOF	Malicious	0.38128	0.99730	0.55166	$k\_neighbors = 3100$
		Benign	0.99982	0.90179	0.94280	
	Autoencoder	Malicious	0.87728	0.99730	0.99730	$T = 93.5\%$
		Benign	0.99983	0.99153	0.99567	
Standard Tailsitter	OC-SVM	Malicious	0.56739	0.99102	0.72162	$\nu = 0.19$
		Benign	0.99941	0.95307	0.97569	$\gamma = 0.000211$
	LOF	Malicious	0.48570	0.99191	0.65210	$k\_neighbors = 3100$
		Benign	0.99946	0.93477	0.96603	
	Autoencoder	Malicious	0.87728	0.99730	0.93345	$T = 93.5\%$
		Benign	0.99983	0.99153	0.99567	
Standard Plane	OC-SVM	Malicious	0.25993	0.99242	0.41196	$\nu = 0.1811$
		Benign	0.99960	0.87150	0.93117	$\gamma = 0.000254$
	LOF	Malicious	0.21127	0.99147	0.34832	$k\_neighbors = 3500$
		Benign	0.99953	0.83167	0.90791	
	Autoencoder	Malicious	0.86068	0.98957	0.92063	$T = 95\%$
		Benign	0.99952	0.99271	0.99611	

[7] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. The MIT Press. 507 pages.

[8] Andrew J Kerns, Daniel P Shepard, Jahshan A Bhatti, and Todd E Humphreys. 2014. Unmanned aircraft capture and control via GPS spoofing. *Journal of Field Robotics* 31, 4 (2014), 617–636.

[9] Samir Khan, Chun Fui Liew, Takehisa Yairi, and Richard McWilliam. 2019. Unsupervised anomaly detection in unmanned aerial vehicles. *Applied Soft Computing* 83 (2019), 105650.

[10] Robert Mitchell and Ray Chen. 2013. Adaptive intrusion detection of malicious unmanned air vehicles using behavior rule specifications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 44, 5 (2013), 593–604.

[11] Devaprakash Muniraj and Mazen Farhood. 2017. A framework for detection of sensor attacks on small unmanned aircraft systems. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 1189–1198.

[12] Hichem Sedjelmaci, Sidi Mohammed Senouci, and Nirwan Ansari. 2016. Intrusion detection and ejection framework against lethal attacks in UAV-aided networks: A Bayesian game-theoretic methodology. *IEEE Transactions on Intelligent Transportation Systems* 18, 5 (2016), 1143–1153.

[13] Hichem Sedjelmaci, Sidi Mohammed Senouci, and Nirwan Ansari. 2017. A hierarchical detection and response system to enhance security against lethal cyber-attacks in UAV networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 48, 9 (2017), 1594–1606.

[14] Jianguo Sun, Wenshan Wang, Qingan Da, Liang Kou, Guodong Zhao, Liguo Zhang, and Qilong Han. 2018. An Intrusion Detection Based on Bayesian Game Theory for UAV Network. In *11th EAI International Conference on Mobile Multimedia Communications*. European Alliance for Innovation (EAI), 56.

[15] Nils Ole Tippenhauer, Christina Pöpper, Kasper Bonne Rasmussen, and Srdjan Capkun. 2011. On the requirements for successful GPS spoofing attacks. In *Proceedings of the 18th ACM conference on Computer and communications security*. 75–86.

[16] Dinghua Wang and Dongqin Feng. 2018. Research on the Strategy of Drones Intrusion Detection Based on Game Theory. In *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE, 613–619.

[17] Jason Whelan, Abdulaziz Almehmadi, Jason Braverman, and Khalil El-Khatib. 2020. Threat Analysis of a Long Range Autonomous Unmanned Aerial System. In *2020 International Conference on Computing and Information Technology (ICCIT-1441)*. IEEE, 230–234.