Deep Learning Detection of GPS Spoofing

Olivia Jullian[1], Beatriz Otero ✉[1], Mirjana Stojilović[2], Juan José Costa[1], Javier Verdú[1], and Manuel Alejandro Pajuelo[1]

[1] Universitat Politècnica de Catalunya, Barcelona, Spain
{ojulian, botero, jcosta, jverdu, mpajuelo}@ac.upc.edu
[2] EPFL, Lausanne, Switzerland
mirjana.stojilovic@epfl.ch

Abstract. Unmanned aerial vehicles (UAVs) are widely deployed in air navigation, where numerous applications use them for safety-of-life and positioning, navigation, and timing tasks. Consequently, GPS spoofing attacks are more and more frequent. The aim of this work is to enhance GPS systems of UAVs, by providing the ability of detecting and preventing spoofing attacks. The proposed solution is based on a multilayer perceptron neural network, which processes the flight parameters and the GPS signals to generate alarms signalling GPS spoofing attacks. The obtained accuracy lies between 83.23% for TEXBAT dataset and 99.93% for MAVLINK dataset.

Keywords: Deep Learning · Intrusion Detection Model · Unmanned Aerial Vehicles · Spoofing · Global Navigation Satellite System.

1 Introduction

Smart devices, such as unmanned aerial vehicles (UAVs), are widely deployed in our society. For every mission, these devices strongly rely on their communications system [11], which is typically based on the Internet of Things (IoT) networks and GPS channels.

In the UAV world, GPS-based systems face two main threats: *jamming* and *spoofing* attacks [11]. In a jamming attack, the attacker goal is a denial-of-service (DoS), so that the UAV is unable to receive the GPS signal. In a spoofing attack, the attacker creates a replica of the GPS signal and boosts its power, for it to become the positioning reference of the UAV. The increased power affects the correlation between the signals from the GPS and the navigation system. Consequently, once the spoofed signal is sent to the UAV, the latter ignores the real GPS signal [18] and starts drifting from the original path.

During a spoofing attack, the target UAV is unable to immediately detect the drift because, if the attack is executed well, there are no abrupt changes in the received GPS signal strength. Additionally, there is no knowledge of the correct position to help the UAV notice the drift. For these reasons, the spoofing attacks are hard to detect.

Research studies on the detection and prevention of spoofing attacks are suggesting that deep neural networks (DNNs) have great potential. However,

their suitability is not fully understood nor comprehensively verified on publicly-available spoofing-attack datasets.

The main contributions of this work are:

- design and software implementation of a multilayer perceptron (MLP) and a long short-term memory (LSTM) DNN for the spoofing-attack detection,
- comparison of their accuracies on MAVLINK and TEXBAT datasets,
- selection of the best DNN for the intrusion detection (ID) framework,
- retraining of the designed models for an additional function—prevention of spoofing attacks (by generating an early alarm for the UAV before the spoofing attack starts), and
- comparison to other machine-learning/deep-learning solutions proposed in the literature.

In the remainder of this paper, Section 2 describes the spoofing attacks. Section 3 presents the approaches proposed. Section 4 describes the related work. Section 5 presents the intrusion detection framework. Section 6 discusses the DNN model design and training. Then, Section 7 gives the experimental results. Finally, the conclusions are given in Section 8.

2 GPS Spoofing Attacks

Communication between the GPS satellites and the UAVs is needed to obtain the flight path of a UAV and for navigation. UAVs use at least four satellites to navigate. Moreover, GPS satellites provide the position reference to the UAVs [20]. Sensors such as inertial measurement units, magnetometers, and gyroscopes are often deployed, for increased precision and security [20].

During navigation, a malicious signal can become the reference for the UAV, even though it is not generated by a GPS satellite. The presence of such a signal defines a spoofing attack. Merwe et al. [9] presented a classification of spoofing attacks considering various aspects, such as synchronization between the original and the spoofed signal, the number of antennas required to attack the vehicle, or the spoofed signal generation.

In an asynchronous attack, the attacker does not monitor the reference GPS signal of the target. Creating a spoofed signal without the knowledge of the reference entails differences in signal characteristics and, simply, a different position being sent to the UAV. Thus, in an asynchronous attack, abrupt position changes are communicated to the UAV, making these attacks easier to detect than synchronous attacks [8].

In a synchronous spoofing attack [8], the attacker tracks the target UAV. Therefore, it knows the target's exact location, which allows the attacker to receive the corresponding reference GPS signal. The attacker creates a spoofed signal by replicating the reference and slightly increasing the power of this new signal. The spoofed signal, once sent back to the target, becomes the new reference, precisely because of its higher power. Additionally, the attacker becomes able to relocate the target by changing the reference signal characteristics.

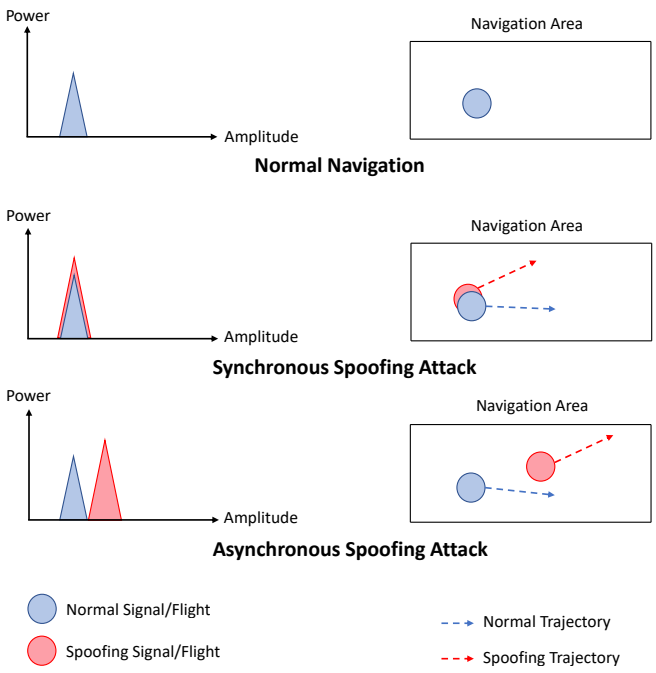

Fig. 1. Synchronous versus asynchronous spoofing attacks.

Figure 1 illustrates the differences between an asynchronous and a synchronous spoofing attack. In this figure, the synchronous spoofing signal is aligned with the reference signal, which is why the target can start drifting without being aware of the attack. On the other hand, in an asynchronous attack, the spoofing signal is not synchronized, which causes an abrupt change in the target's position and makes the attack easier to detect.

For the above mentioned and other limitations of asynchronous spoofing attacks (e.g., listed by Merwe et al. [9]), this work focuses on detection and prevention of the more serious threats: synchronous spoofing attacks [8].

3 Proposed Approaches

After presenting the threat of spoofing attacks and understanding their impact on the UAVs, we present here our strategies for the detection and prevention of synchronous spoofing attacks (illustrated in Figure 2).

3.1 Intrusion Detection Framework for Spoofing Attack Detection

First, we propose to design an intrusion detection (ID) software framework for detecting GPS spoofing attacks. The detection will be done using only the flight

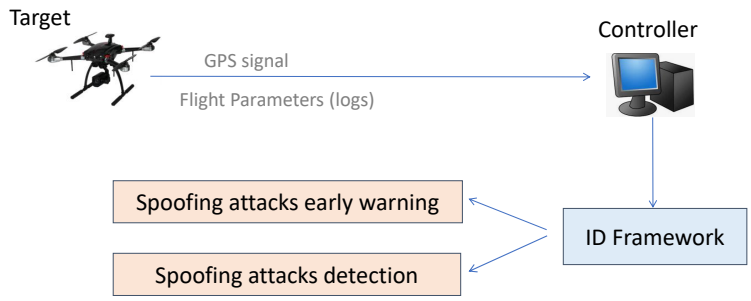

Fig. 2. Illustration of our proposed solution for synchronous spoofing attack detection and prevention; it involves a flight controller, a communication channel between the target and the controller, and a software intrusion detection framework (DNN-based).

parameters collected during one measurement cycle (e.g., one clock cycle). Communication between the flight controller and the target UAV device serves to offload the flight parameters and the GPS signal characteristics from the device. The flight controller then analyzes the received data and, with the help of the ID framework, detects the attack and signals an alarm.

3.2 Early-Warning Alarm for Spoofing Attack Prevention

In the same line of research, another approach is considered here: The ID framework described previously should also be able to generate an alarm for warning that a spoofing attack may be taking place. To that purpose, the flight parameters collected during several subsequent measurement cycles need to be processed (e.g., GPS signal power, GPS signal amplitude, changes in the UAV orientation, system status changes, etc.).

4 Related Work

4.1 Techniques for GPS Spoofing Attack Detection

In the last decade, a number of research studies focused on aircraft and navigation security problems. In the recent couple of years, UAVs have become increasingly popular and, consequently, their security vulnerabilities. Most common threats to UAVs rely on the IoT protocols or on the GPS communication.

M. P. Arthur [1] categorizes the UAV threats in three types: navigation attacks (hijacking), routing attacks (based on the IoT network), and data attacks (where data is stolen from hijacked drones). Related to navigation attacks, jamming and spoofing attacks are identified as the main threats.

Several works address detection of the spoofing attacks. Morales-Ferre et al. [10] use the Monte Carlo approach to compare two detectors: the sum of squares detector and the D^3 detector. E. Shafie et al. [18] take a more traditional approach: They test models such as a Bayesian classifier and the K-Nearest

Neighbour (K-NN) classifier, achieving accuracies of 62.31% and 77.29%, respectively, when detecting a synchronous spoofing attack.

Ranganathan et al. [16] develop SPREE, an approach for synchronous spoofing attacks detection. SPREE is a software-radio solution based on tracking the reference as well as the auxiliary GPS signals, by allocating more than one channel to the same satellite.

In the next section, we survey the use of deep learning (DL) techniques and identify the most promising candidates for our intrusion detection framework.

4.2 Deep Learning for GPS Spoofing Attack Detection

S. Semanjski et al. [17] used support vector machines (SVMs) to detect spoofed signals while lifeguard systems are flying. Despite the improvement in accuracy compared to non-ML-based techniques, in the same research [17] the authors conclude that the ML methods (in particular, SVM) are not sufficient. The reason lies in the nonlinear characteristics of the attacks. To deal with nonlinear data, the authors use kernels SVM (to transform nonlinear attack characteristics to linear ones through mathematical algorithms)—they attach many kernels to an SVM detector, achieving 94.41% accuracy.

Unsupervised DL techniques are also used to create datasets from the UAV sensor readings. It is the case of the work of M.P. Arthur et al.[1], where self-taught learning was used to develop a new dataset. An SVM model is used in the previous dataset with an accuracy of 94% with a framework installed in the UAV.

E. Shafie et al. applied an MLP model to detect spoofing attacks with a 99.3% accuracy [18]. Another DL approach [12] used convolutional neural networks (CNNs), achieving 94% to 99% accuracy.

In the research of G. Bae et al. [2], a distributed DL framework is presented, which reduces the training time per epoch from 30 to 5 seconds using an autoencoder-based LSTM model. In the work by K. H. Park et al. [14], an autoencoder (AE) is used to predict spoofing attacks during a flight.

In Table 1, we summarize the results of the previous studies on synchronous spoofing attack detection using deep learning techniques. The table compares MLP, C-SVM, LSTM-AE, self-taught learning, and AE-based approaches. Their accuracies range between 93.4% and 99%. An important drawback of these studies is the use of synthetic datasets and the lack of comparison to other DL models. Only one general dataset (based on simulated flight parameters) was used to test spoofing attacks detection [14], but no common metrics for comparing to other studies were considered. Hence, a comprehensive comparison of various DL techniques for detecting GPS spoofing attacks using a common dataset is still missing. In this work, we address the above issue by comparing our DNN models to SVM and random forest models, using the same datasets (Section 7).

The lack of a general dataset for spoofing attacks makes it difficult to choose the most suitable framework for spoofing attack detection. However, after comparing the ML and DL approaches presented in previous studies [1][15][18], we can conclude that the DL techniques are most promising. Moreover, they are

Table 1. Survey of synchronous attack detectors based on deep learning models.

Model	Metrics	Dataset
MLP [18]	ACC (99.3%)	Synthetic
C-SVM [17]	ACC (94.41%)	Synthetic
LSTM-AE [2]	ROC/ACC(93.4%)	Sensors data
Self-taught learning [1]	ACC (94%)	Sensors data
AE [14]	Error attack (0.25)/F1-score (94.81%)	MAVLINK

proven suitable not only for detection but also for prediction (and thus prevention) of an attack [14].

5 DL Models In Our Intrusion Detection Framework

5.1 MLP for GPS Spoofing Attack Detection

One of the objectives of this work is the comparison between MLP and LSTM deep-learning models, as an integral step of the design of an ID framework for spoofing attacks detection. The choice of MLP is motivated by the excellent accuracies reported in previous works, reaching 99% in the Meaconer attacks (i.e., denial-of-service attacks).

An MLP model is composed of a number of layers (some of which are hidden), in which the training flow follows one direction only (from the input towards the output layer). Additionally, MLP is a fully-connected neural network, in which the units in subsequent layers are fully interconnected, and each connection is weighted. All the weights are combined together to compute the output of a unit using activation functions. Figure 3 shows an MLP model composed by four hidden layers. The exact details of the MLP model used in this work are given in Section 6.3.

5.2 LSTM for GPS Spoofing Attack Detection

Besides the MLP model, we develop an LSTM model. We use flight parameters and GPS signal characteristics as time-sequential data.

As the result of the overcoming of the recurrent neural network (RNN) gradient vanishing, LSTM generates the output of a unit using recurrently the connections among hidden units. Three gates take the control of the information flow: the input, the forget, and the output gate. These three gates allow the unit to calculate its state (using the input provided by the input gate). They also allow considering previous information from other units (decided by the forget gate). Finally, in an LSTM model, after the output of the unit is computed, the output gate decides whether or not to consider the previous information and transmits its decision to the unit through the forget gate. Figure 4 illustrates an LSTM model with three hidden layers. The exact details of the LSTM model used in this work are given in Section 6.3.

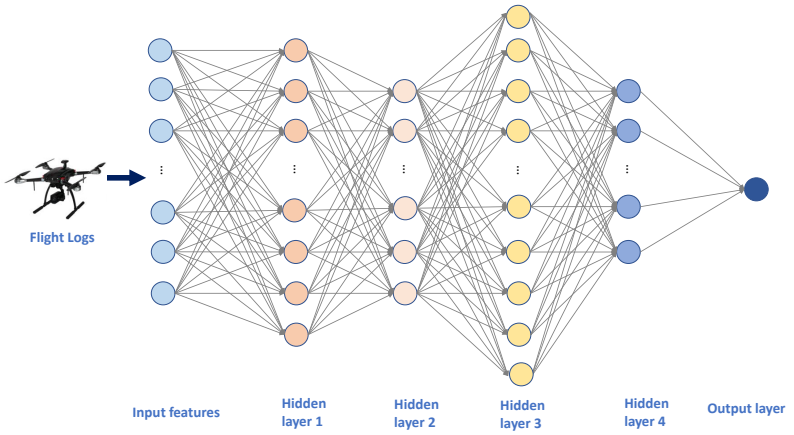

Fig. 3. An MLP for GPS spoofing attack detection.

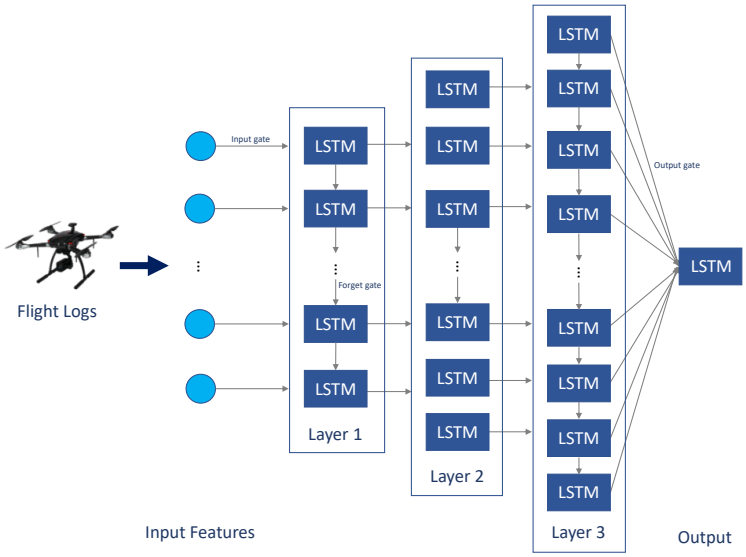

Fig. 4. An LSTM model for GPS spoofing attack detection.

6 Model Design and Training

To train and test the two DNN models presented in Section 5, we choose two datasets: Texas Spoofing Test Battery (i.e., TEXBAT [5]) and MAVLINK [19]. The comparison results will guide us in choosing better of the two models for the ID framework.

6.1 Datasets Description

MAVLINK Dataset. This dataset is composed of flight system parameters (also called flight logs) collected using PX4 autopilot and Gazebo robotics simulator [19]. PX4 is an open-source autopilot firmware used in many UAVs. Therefore, MAVLINK dataset contains *general* data—data corresponding to a large number of UAVs [19]. The dataset contains two groups of data samples: The first type corresponds to the parameters of a routine flight (in the absence of the attack). The second type corresponds to the parameters of a flight under a spoofing attack (also called a spoofed flight). It should be noted that the spoofing attack lasts 30 seconds, while the flight takes 10–30 minutes; in other words, this dataset is imbalanced.

A large number of features with little variance are present in the dataset. This lack of variance can make the model focus on features with no relevance for the problem at hand (even when applying the principal component analysis technique). To avoid this scenario, we consider only the flight parameters with high variance:

- GPS coordinates: latitude, longitude, military grid reference system, and the course over the ground.
- Position and orientation logs: relative altitude, roll, pitch, heading, roll rate, pitch rate, yaw rate, and ground speed.
- System and control status: air speed, climb rate, distance to home, next GPS signal transmitter to visit, throttle, and battery measurements.

Many MAVLINK features relate to the position and the location, which is convenient for the spoofing attack detection—due to the drift caused by the spoofing attack, orientation, system, and control status features are affected. Additionally, GPS signal characteristics contain information, as power and phase changes in the reference GPS signal.

TEXBAT Dataset. TEXBAT [5] is a publicly available dataset commonly used for testing the resilience of GPS receivers. It contains digital recordings of live static and dynamic GPS L1 C/A spoofing tests. The characteristics of TEXBAT allow the dataset to represent a generalisation of the spoofing attacks detection problem, where not only UAVs are considered but any vehicle with a GPS.

Among all the spoofing attacks covered by TEXBAT dataset [4][5], DS3 and DS7 scenarios have the characteristics of a synchronous spoofing attack. The DS3 scenario is based on static matched-power time push attacks. The DS7 scenario explores the same spoofing attack as DS3, while employing carrier phase alignment. Since DS7 is based on the DS3 scenario and, by introducing a new alignment the complexity of intrusion detection is increased, we focus on the DS7 scenario.

The TEXBAT dataset contains binary data. Using GRID code [6], we convert the binary into the navigation data, which we then use to train the DNN models in this work. In future work, we consider using the navigation data for real-time processing (by a digital signal processor or a field-programmable gate array).

It should be noted that, given that all the data in TEXBAT dataset corresponds to spoofed flights (i.e., the data describing regular flights is not provided), we use TEXBAT to train for and test the attack detection only (Section 3).

6.2 Datasets Preprocessing

This section focuses on the preprocessing of the chosen datasets. MAVLINK dataset is a set of flight parameters (i.e., flight logs) extracted from an autopilot simulator. The difference between TEXBAT and MAVLINK dataset is that the data in the TEBXAT dataset can be given to the controller without any modification, because they represent physical features of the GPS signals. On the contrary, MAVLINK data (flight logs) contain very many features (some highly correlated) and thus require preprocessing.

Dataset balancing. The DNN models in this work will be designed with two scenarios in mind: first, for spoofing attack detection and, second, for spoofing attack prevention (early-warning alarm). In the first case, the training dataset composed of samples of a spoofed flight only is used; this dataset is imbalanced, as the attack duration is considerably shorter than the duration of the entire flight. The exact number of attack data samples in both TEXBAT and MAVLINK dataset is shown in Table 2 (80% for training and 20% for testing). In the second case, the training dataset is composed of equal number of flight logs (data samples) from a regular flight and logs from a spoofed flight. Therefore, the training dataset is balanced (because both the regular and the spoofed flights last equal time). The DNN model is built using the latter dataset, while evaluated on both.

Table 2. Dataset (im)balance.

	MAVLINK		**TEXBAT**	
	Train	**Test**	**Train**	**Test**
Attack samples	187	7	39	6
Normal flight samples	5728	1433	26149	6542

To balance the data, oversampling techniques can be used. We choose synthetic minority oversampling technique (SMOTE), which uses interpolation to create new samples. Thus, after oversampling spoofing attacks samples, we obtain the same number of attack samples and the normal flight samples and, consequently, ensure abstraction and prevent overfitting.

MAVLINK dataset preprocessing. Different techniques are required before training the models, to prevent overfitting problems. First, all the MAVLINK

spoofing datasets originating from various UAVs are joined, to have as many samples as possible. Then, one-hot encoding is applied to treat categorical values. Finally, principal component analysis (PCA), a technique based on the correlation between the features, is used to reduce the number of input features [3]. We chose ten input features.

6.3 Models Structure

Hyperband [7] is used to tune the DNN models for both scenarios (detection and prevention) and for both datasets (MAVLINK and TEXBAT).

MAVLINK Dataset Models. The MLP model is set up and trained with four hidden layers (the first with 96 units and Sigmoid function and the remaining layers with 76, 118, and 36 units, respectively, using rectified linear unit (ReLU) activation function). A dropout rate of 0.75 is used at the end, to avoid overfitting. The output layer is a single perceptron layer with a Sigmoid activation function, which enables the output to be limited between 0 and 1. We use binary cross-entropy loss function and the Adam optimizer, with a learning rate of 0.001, running for at most 30 epochs. The batch size has 1,900 samples.

On the other hand, the LSTM model has three hidden layers with 130, 132, and 164 units, respectively. The batch size is 1,000 samples and there are at most 10 epochs. The LSTM DNN uses the same optimizer, loss function, and the learning rate as the MLP model. To improve the accuracy, when training the model in the early-warning alarm prediction, the number of epochs is increased empirically to 76 for the MLP model and to 100 for the LSTM model.

TEXBAT Dataset Models.
The MLP model for spoofing attack detection based on the TEXBAT dataset is trained with three hidden layers (with 102, 78, and 70 units, respectively, using ReLU activation function). The output layer is a single perceptron unit with a Sigmoid activation function. We use binary cross-entropy loss function and the Adam optimizer, with a learning rate of 0.001, running for at most 12 epochs. The batch size has 100 samples.

The LSTM model has two hidden layers with 132 and 164 units, respectively. The batch size has 1,000 samples, running for a maximum of 12 epochs. The LSTM model uses the same optimizer, loss function and learning rate as the MLP model.

7 Results

For the intrusion detection framework design, two experiments using flight parameters (MAVLINK dataset) are considered: one for detecting the spoofing attacks and one for predicting them (raising an early-warning alarm). Table 3 summarizes the results obtained for each experiment. Similarly, Table 4 shows

Table 3. Results for attacks detection and prevention on the MAVLINK dataset.

	Attack detection		Attack prevention	
Metrics	**MLP**	**LSTM**	**MLP**	**LSTM**
ACC	99.93%	99.93%	94.43%	85.93%
Precision	99.96%	100.0%	94.94%	77.84%
Recall	85.71%	85.71%	100.0%	100.0%
F1-Score	92.29%	92.31%	97.41%	87.54%

Table 4. Results for attack detection on the TEXBAT dataset.

DL Model	**ACC**	**Precision**	**Recall**	**F1-Score**
MLP	83.23%	87.07%	67.14%	82.79%
LSTM	82.1%	91.04%	75.58%	82.59%

the results of the spoofing attack detection on the TEXBAT dataset. In our experiments, 80% of the data is used for training, while the remaining 20% is used for validation. Clearly, DNNs provide excellent results, confirming the ability of MLP and LSTM to detect spoofing attacks, even though the two models consider the same scenarios in different ways: The LSTM model takes into account the past model inferences while MLP considers only the current units' inferences.

It can be observed that LSTM models have a slightly lower accuracy than MLP. This difference is linked to the very definition of a spoofing attack: If the target is considering previous flight records when a spoofing attack is taking place, then a spoofing attack will be only detected once the target is drifting to an undesired position (due to the similarities between the previous samples and the actual attack samples). For LSTM models, the fact that the spoofing attack signals are at first almost identical to the previous GPS signals has a negative impact on the model. This impact is translated into a delay of attack detection compared to MLP models.This negative impact is demonstrated in Table 3, where MLP have better F1-Score than LSTM models. Also the same behavior is observed when using MLP and LSTM models for general spoofing attacks ID (Table 4), with a difference between MLP and LSTM F1-Score of 0.2%.

On the other hand, and diving into MALVINK dataset results, it is also logical to think that raising an early alarm for spoofing attacks will be harder than detecting the threats. The results reflect this logical statement, since the accuracy reduces from 99.93% to 94.43% in MLP models and from 99.93% to 85.93% in LSTM models. Even with this difference, the same DNN has demonstrated that it has a great ability (more than 85% of the cases for the MAVLINK dataset) to not only detect spoofing attacks, but also warn if the UAV is under threats (with accuracies of 94.43% for MLP models and 85.93% for LSTM models). We can therefore conclude that MLP is a better solution for our intrusion detection framework.

Table 5. ML/DL solutions for spoofing attacks ID

Detection model	Metrics	Dataset
AE [14]	F1-score (94.81%)	MAVLINK (detection)
Random forest	F1-score (89.21%)/ACC (89.33%)	MAVLINK (detection)
SVM	F1-score (95.99%)/ACC (96%)	MAVLINK (detection)
MLP	**F1-score (**92.29%**)/ACC (**99.93%**)**	MAVLINK (detection)
Random forest	F1-score (66.68%)/ACC (68.04%)	MAVLINK (warning)
SVM	F1-score (86.18%)/ACC (86.22%)	MAVLINK (warning)
MLP	**F1-score (**97.41%**)/ACC (**94.43%**)**	MAVLINK (warning)
Random forest	F1-score (47.52%)/ACC (56.77%)	TEXBAT (detection)
SVM	F1-score (83.25%)/ACC (82.3%)	TEXBAT (detection)
MLP	**F1-score (**82.79%**)/ACC (**83.23%**)**	TEXBAT (detection)

Finally, we provide a comparison between our MLP solution and the DL techniques presented in previous studies. The comparison is performed using the same MAVLINK and TEXBAT datasets. Obtained results are summarized in Table 5. These results further confirm that the MLP solution developed in this work is not only suitable for detecting GPS spoofing attacks but also superior to a number of other approaches.

8 Conclusions

The number of applications of unmanned aerial vehicles is rapidly growing and, with it, GPS spoofing attacks are becoming a serious threat. In this work, a software-based intrusion detection framework capable of not only detecting but also predicting the attack (early-warning system) was developed. The framework is based on a DNN, trained and verified on two datasets: MAVLINK and TEXBAT. Two DNN models were compared: MLP and LSTM, with the experimental results showing that MLP is superior. On MAVLINK dataset, accuracies between 99.43% and 99.93% were achieved, while on TEXBAT dataset the accuracy reached 83.23%. Finally, the resulting MLP model was compared with two other DL-based approaches presented in the literature—random forest and SVM—and demonstrated to be better-performing.

The evolution of DNNs shows that they can significantly improve the classification accuracies in many applications [13]. However, a well-performing DNN is often large: It requires a high number of units, hidden layers, or features. This translates to a continuous increment in computational requirements, memory bandwidth, and storage needed to save the model and move the data. For UAVs, these requirements are difficult to satisfy. Hence, UAVs require a trade-off between the model complexity and the computational and storage resources. For that reason, as part of the future work, we will focus on reducing the complexity of the attack detection model and making it feasible for the model to reside and execute on the UAV.

Acknowledgments. This work was supported in part by the Catalan Government, through the program 2017-SGR-962 and the RIS3CAT DRAC project 001-P-001723, and by the EPFL, Switzerland.

References

1. Arthur, M.P.: Detecting signal spoofing and jamming attacks in UAV networks using a lightweight IDS. In: 2019 International Conference on Computer, Information and Telecommunication Systems (CITS). pp. 1–5 (2019). https://doi.org/10.1109/CITS.2019.8862148
2. Bae, G., Joe, I.: UAV anomaly detection with distributed artificial intelligence based on lstm-ae and ae. In: Yang, L., Hao, F., Jeong, Y.S., Park, J. (eds.) Advanced Multimedia and Ubiquitous Engineering - MUE/FutureTech 2019. pp. 305–310. Lecture Notes in Electrical Engineering, Springer Verlag (2020). https://doi.org/10.1007/978-981-32-9244-4_43
3. Holland, S.M.: Principal components analysis (PCA). Department of Geology, University of Georgia, Athens, GA pp. 30602–2501 (2008)
4. Humphreys, T.: TEXBAT data sets 7 and 8. The University of Texas (2016)
5. Humphreys, T.E., Bhatti, J.A., Shepard, D., Wesson, K.: The texas spoofing test battery: Toward a standard for evaluating GPS signal authentication techniques. In: Radionavigation Laboratory Conference Proceedings (2012)
6. Joplin, A., Lightsey, E.G., Humphreys, T.E.: Development and testing of a miniaturized, dual-frequency GPS receiver for space applications. In: Institute of Navigation International Technical Meeting, Newport Beach, CA (2012)
7. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A.: Hyperband: A novel bandit-based approach to hyperparameter optimization. Journal of Machine Learning Research **18**(1), 1–52 (2017)
8. Mendes, D., Ivaki, N., Madeira, H.: Effects of GPS spoofing on unmanned aerial vehicles. In: 2018 IEEE 23rd Pacific Rim International Symposium on Dependable Computing (PRDC). pp. 155–160 (2018). https://doi.org/10.1109/PRDC.2018.00026
9. v. d. Merwe, J.R., Zubizarreta, X., Lukčin, I., Rügamer, A., Felber, W.: Classification of spoofing attack types. In: 2018 European Navigation Conference (ENC). pp. 91–99 (2018). https://doi.org/10.1109/EURONAV.2018.8433227
10. Morales-Ferre, R., Richter, P., Falletti, E., de la Fuente, A., Lohan, E.S.: A survey on coping with intentional interference in satellite navigation for manned and unmanned aircraft. IEEE Communications Surveys Tutorials **22**(1), 249–291 (2020). https://doi.org/10.1109/COMST.2019.2949178
11. Mozaffari, M., Saad, W., Bennis, M., Debbah, M.: Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs. IEEE Transactions on Wireless Communications **15**(6), 3949–3963 (2016). https://doi.org/10.1109/TWC.2016.2531652
12. Munin, E., Blais, A., Couellan, N.: Convolutional neural network for multipath detection in gnss receivers. In: 2020 International Conference on Artificial Intelligence and Data Analytics for Air Transportation (AIDA-AT). pp. 1–10 (2020). https://doi.org/10.1109/AIDA-AT48540.2020.9049188
13. Nurvitadhi, E., Venkatesh, G., Sim, J., Marr, D., Huang, R., Ong Gee Hock, J., Liew, Y.T., Srivatsan, K., Moss, D., Subhaschandra, S., Boudoukh, G.: Can FPGAs beat GPUs in accelerating next-generation deep neural networks?

In: Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. p. 5–14. FPGA '17, Association for Computing Machinery, New York, NY, USA (2017). https://doi.org/10.1145/3020078.3021740, https://doi.org/10.1145/3020078.3021740
14. Park, K.H., Park, E., Kim, H.K.: Unsupervised intrusion detection system for unmanned aerial vehicle with less labeling effort. In: You, I. (ed.) Information Security Applications. pp. 45–58. Springer International Publishing (2020). https://doi.org/10.1007/978-3-030-65299-9-4
15. Quan, Y., Lau, L., Roberts, G.W., Meng, X., Zhang, C.: Convolutional neural network based multipath detection method for static and kinematic GPS high precision positioning. Remote Sensing **10**(12) (2018). https://doi.org/10.3390/rs10122052
16. Ranganathan, A., Ólafsdóttir, H., Capkun, S.: Spree: A spoofing resistant GPS receiver. In: Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking. pp. 348–360 (2016)
17. Semanjski, S., Semanjski, I., De Wilde, W., Muls, A.: Cyber-threats analytics for detection of GNSS spoofing. In: DATA ANALYTICS 2018: The Seventh International Conference on Data Analytics. pp. 136–140. IARIA (2018)
18. Shafiee, E., Mosavi, M.R., Moazedi, M.: Detection of spoofing attack using machine learning based on multi-layer neural network in single-frequency GPS receivers. Journal of Navigation **71**(1), 169–188 (2018). https://doi.org/10.1017/S0373463317000558
19. Whelan, J., Sangarapillai, T., Minawi, O., Almehmadi, A., El-Khatib, K.: UAV attack dataset (2020). https://doi.org/10.21227/00dg-0d12, https://dx.doi.org/10.21227/00dg-0d12
20. Zhi, Y., Fu, Z., Sun, X., Yu, J.: Security and privacy issues of UAV: A survey. Mobile Networks and Applications pp. 95–101 (2020). https://doi.org/10.1007/s11036-018-1193-x