engineering proceedings

MDPI

Proceeding Paper

Antiference: New Concept for Evolutive Mitigation of RFI to GNSS †

Shahrzad Afroozeh [1,*], Vincent Bejach [2], Uros Bokan [1], André Bos [2], Bastiaan Ober [3] and Sascha Bartl [1]

[1] OHB Digital Solutions GmbH, Kärntner Straße 7b/1, 8020 Graz, Austria; uros.bokan@ohb-digital.at (U.B.); sascha.bartl@ohb-digital.at (S.B.)
[2] S&T—Science and Technology BV, Olof Palmestraat 14, 2616 Delft, The Netherlands; vincent.bejach@stcorp.nl (V.B.); andre.bos@stcorp.nl (A.B.)
[3] Integricom, Tjalkenwerf 30, 2317 Leiden, The Netherlands
* Correspondence: shahrzad.afroozeh@ohb-digital.at
† Presented at the European Navigation Conference 2023, Noordwijk, The Netherlands, 31 May–2 June 2023.

Abstract: The past decade has shown a growing awareness of the dangers of intentional interference (especially jamming and spoofing) with GNSS signals. The Antiference project uses reconfigurable digital signal processing methods in the detection, classification, and mitigation of interference by employing machine learning techniques. The ML-based jamming classifier uses distinctive features of spectrograms for the differentiation of various jamming attacks. A residual neural net is used to map the spectrograms to the different jamming types. It relies on a fingerprinting architecture. Fingerprints summarize the characteristics of all the incoming signals, which are stored in and matched to a database of previously encountered interference types. To validate the implemented functionalities, a developed test-bed runs test scenarios and benchmarks the results against two state-of-the-art COTS receivers with interference mitigation capabilities.

Keywords: machine learning; jamming; spoofing; ResNet50; COTS receivers; interference detection and mitigation

Citation: Afroozeh, S.; Bejach, V.; Bokan, U.; Bos, A.; Ober, B.; Bartl, S. Antiference: New Concept for Evolutive Mitigation of RFI to GNSS. *Eng. Proc.* **2023**, *54*, 61. https://doi.org/10.3390/ENC2023-15451

Academic Editors: Tom Willems and Okko Bleeker

Published: 29 October 2023

1. Introduction

The past decade has shown a growing awareness in the dangers of intentional interference in the area of GNSS signals. Jamming and Spoofing (the two main intentional and harmful radio frequency interference (RFI) types) have been receiving a lot of attention in terms of the availability and affordability of hardware to induce the attacks, and the resulting costs to the victims. Jamming is the broadcasting of a radio frequency (RF) signal with the goal of blocking the reception of authentic GNSS signals and it leads to the complete blockage of the service or to the significant degradation of the service quality. Victims of jamming usually know that they are experiencing a form of interference. However, newer jamming methods, such as low-power jamming, code/spectrum-matching jamming (or PRN jammers) [1], and systemic jamming [2] lead to the degradation or blocking of the service without the victim being completely aware of what is happening.

Spoofing, on the other hand, is the targeted broadcasting of counterfeit GNSS signals so that the victim receiver will perceive them as the authentic signals and follow the counterfeit signals. In these types of attacks, the victim receivers are not aware of being attacked and can be steered to unknown locations or be prone to accidents.

The Antiference project uses reconfigurable digital signal processing (DSP) methods and tools in the detection, classification, characterization, and mitigation of harmful RFI. While some forms of RFI signal mitigation do not rely on the detection and classification of the RFI (like Notch filters or narrow correlators for multipath mitigation), effective mitigation in most cases relies on knowing the parameters of the interfering signal and can be significantly improved if accurate knowledge about the type and parameters of the attack is known. Employing data-driven artificial intelligence (AI) and machine learning

(ML) techniques to facilitate the detection and classification of RFI is the main focus of this project. However, like many other defensive procedures, the recording and sharing of previously encountered attacks (RFI incidents in this case) can save time in their detection and classification and facilitate effective mitigation measures. Since ML models can suffer from explicit knowledge representation and storing that can be shared between multiple equipment (multiple instances of the same model), a fingerprinting (FP) architecture is introduced to alleviate the possible shortcomings. The FPs of each incoming signal have been generated and matched to a database that contains all the previously encountered RFI types. This results in a database of fingerprints that can be shared between devices of the possible RFIs. To test and validate the implemented functionalities, a testbed has been developed that can run test scenarios and benchmark the results with state-of-the-art COTS receivers which include RFI mitigation capabilities.

2. Background

The detection of interference has been a topic of research for a long time. Most of the classical ways to detect jamming are based on signal processing or statistical techniques, such as automatic gain control (AGC) monitoring, spectral monitoring, time-domain statistical analysis, and signal/carrier to noise ratio (CNR) monitoring. It is trickier to detect spoofing, but the most common methods use Position/Velocity/Time (PVT) estimate consistency checks, navigation data estimate consistency checks, the monitoring of AGC for anomalies, the monitoring of multiple correlation peaks and distortions of the Autocorrelation function (ACF), using additional equipment like inertial measurement units (IMU) for position check, signal polarization, or multi-antenna arrays for directing arrival detection.

Machine learning (ML)-based methods have also been explored to detect jamming and spoofing signals. In some cases, they use similar features to the classical techniques as a basis for the detection of interference. Some of the features used can be summarized as Power Law Detector (PLD), such as the monitoring of AGC-gain values; Power Spectral Density (PSD) Analysis, including methods such as the computation of Power Spectral Density (PSD), filter banks, and wavelet Transformations; Probability Density Function (PDF) Analysis, such as Kurtosis and the Gaussian White Noise test; Short Term Fourier Transform (STFT) for the detection and classification of jamming; and Cyclo-stationary analysis to detect spoofed conditions, such as pre-correlation Spectral Power Contents Analysis (SPCA) [3].

Using the pre-processed data as features is only part of the solution; the other part is the choice of model and its training and testing data and methods. Jamming detection has been implemented successfully using Support Vector Machines (SVM) in [4–6], while Neural networks (NN) such as Convolutional Neural Networks have been used in [7]. RFI classification has used SVMs [8,9], Long Short-Term Memory (LSTM) has been used in [3], and Convolutional Networks has been used in [8] with good results. Spoofing detection is carried out using SVMs in [10,11], using NNs in [11,12], and using the LASSO technique in [13]. Signal Characterization in the cognitive radio domain is carried out using different NNs [14–18] and SVMs [19].

The mitigation of jamming is performed using multiple classical techniques depending on the types of attack. The most used techniques can be summarized as the Bandpass filter (for out-of-band interfering signals), the Notch Filter (NF) [20], the Adaptive Notch Filter (ANF) [21,22], Frequency Domain Adaptive Filtering (FDAF), Pulse Blanking [20], Wavelet-based methods (such as [23]), and Karhunen–Loeve-based methods [23].

3. Models

The Antiference system uses two NN models for the detection and classification of different RFI types. One model is concerned with the detection and classification of different jammer types, and the other detects the existence of spoofing signals. Figure 1

shows the two models used for jamming detection and classification (left) and spoofing detection (right).

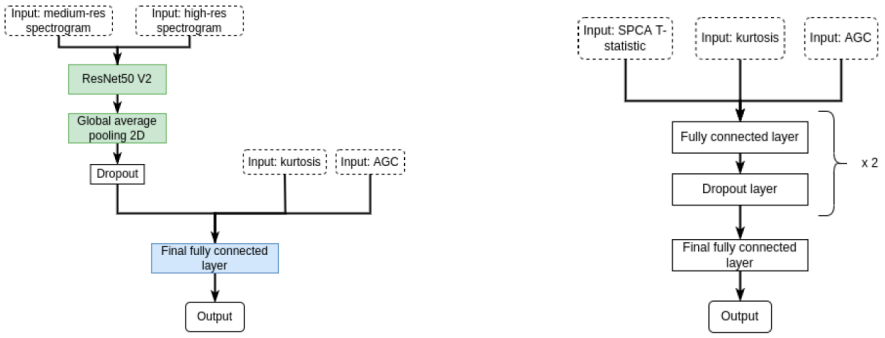

Figure 1. Detection and classification models used in Antiference for jamming (**left**) and spoofing (**right**).

Different features were used for the detection of jamming and spoofing. This implied that separate models would need to be used (otherwise the models would either not be efficient or their size would increase dramatically), which, in turn, also led to having two different FPs and FP databases. The FP of the incoming signals will be created and matched against the FP database for each model separately. If no matches are found, the needed features are extracted from the signal and sent for detection and classification through the NN. The results will be both inserted in the FP database and also sent to the mitigation unit for parameter extraction and mitigation of the signal.

The jamming detection/classification NN is built using transfer learning. It is illustrated in Figure 1, and is composed of two parts: (i) In green: this part extracts features from the spectrograms. Its main component is a pre-trained 50 layers ResNet architecture (see [24] for details about this type of architecture), available through the Keras toolbox (see [25]). The final pooling layer is used to reduce the dimension of the feature map between this step and the decision layer. This reduces the required computations down the network while acting as a "summary" of the most prominent features. In addition, a dropout layer is inserted between the extracted features and the rest of the network to ensure the robustness of the training. (ii) In blue: features are processed by the decision layer, using a fully connected (FC) decision layer. This layer combines the features extracted by the ResNet50 with additional information (the kurtosis and AGC value of the sample).

The ResNet50 V2 architecture was selected by comparing the training speed and classification performance of several architectures commonly used for transfer learning on a subset of the available training data. It is the one that showed the best compromise between the complexity of training and a good prediction of all the classes.

In addition, the parameters of this architecture and of the training process were tuned to enable the highest performance, using a process of Bayesian optimization (see [26,27] for an overview of the tool used to perform the optimization).

4. Data

Since ML-based models require a large number of samples for proper training and assessments, and based on the automotive standards that were used as the basis for assessments of the results, it was decided that the data used for training the model and final validation will use both recordings and simulations to fulfil the requirements.

To do so, three types of data were used: (i) Purely simulated data used for the simulation of all spoofing and PRN jamming datasets; (ii) Recorded base GNSS signal + simulated RFI used for all jamming types other than code/spectrum-matched jammers; and (iii) Recorded signals containing both GNSS + RFI used only for our final validation tests.

It should be noted that the broadcasting of jamming and/or spoofing signals is illegal. All the broadcasting and subsequent recordings of RFI in the real field was carried out

on a military training ground in Austria by OHB Digital Solutions GmbH under a special permit. The difficulty in recording these datasets and their subsequent scarcity is the reason they were used for our final validation. To record the required signals, two data-collection campaigns were performed: firstly, Real-field RFI recordings, collected on the military training ground in Austria within a period of two days. The recordings consist of a collection of jamming and spoofing datasets in static and dynamic conditions. The recorded jammers were five COTS jammers, and broadcasted simulated jamming signals using an IZT S1000 signal generator with a connected power amplifier and transmit antenna. Secondly, Base (clean) GNSS signal recordings collected in the city of Graz and its surrounding areas in Austria. Twelve datasets containing around 8 h of recording in different urban and suburban environments from a car were collected.

Training and Test Data Preparation

To prepare the training data, the first eleven routes from Graz recordings were combined with simulated jamming signals. A few hundred jamming signals were created using different parameters, and combined with chunks of 10 ms of the recorded (base) GNSS signals. The signal combinations were mix-and-match to remove bias in the learning results of the model.

All the real-field RFI recordings, route 12 of the drive-throughs in the city of Graz, and a selection of purely simulated data were used for final validation.

5. Antiference Testbed

The testbed is designed and configured to facilitate the verification and validation of the Antiference system as well as for the benchmarking of the results in relation to the two COTS receivers which have state-of-the-art RFI mitigation techniques implemented in them. Figure 2 shows the Antiference testbed architecture. As can be seen in the image, the COTS receivers can be used to process the incoming signals with or without their internal mitigation mechanisms tuned on, or to process the mitigated signal using the Antiference signal. The testbed has the following sections:

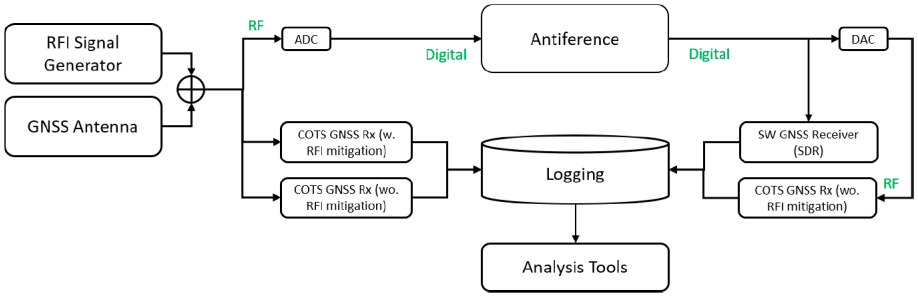

Figure 2. Antiference testbed architecture.

Signal simulator: The signal simulator can be used for the simulation of clean, clean and RFI, or only RFI signals in the GNSS frequency bands. In this project, OHB's GNSS simulator XPLORA was used for the simulation of all simulated signals. Analogue signals recorded through the GNSS antenna can be directly fed into the COTS receivers (to be directly processed by them) or be converted to digital signals to be fed into the Antiference system. Antiference only works in the digital domain. Analogue to Digital Convertor (ADC): in this project, a GNSS frontend based on a 8-bit ADC was used (TeleOrbit MGSE). Two COTS receivers are used for the benchmarking of the results: (1) a low-cost receiver (Rec01) and (2) a mid-range receiver (Rec02) suitable for the automotive domain. SDR (software defined receiver): OHB Digital Solutions' GNSS SDR was used in this project, to process the results of the Antiference, and verify if any residual RFI could be detected using the built-in RFI detectors. Digital to Analogue Convertor (DAC): For the conversion of digital to analogue signals, the built-in convertor in TeleOrbits' MGSE (Multi-GNSS

Simulation and Test Environment) was used for the broadcasting of digital signals for processing using the COTS receivers. The logging manager (or the collection of logs created by each of the elements): to be stored for analysis, signal by signal or in batches. Analysis tools: developed in MATLAB, these are used to analyze the results for the final validation and assessment of the performance

6. Test and Validation Results

We have seen good detection capabilities for both jamming and spoofing. The classification of individual signal chunks has yielded good results, and has proven to provide useful information for the mitigation stage. Table 1 summarizes some of the results.

Table 1. Detection accuracy of the Antiference system.

Datasets	AM	FM	SCW	CW	AGW
False Alarm Rate	0.001	0.002	0.002	0.0	0.002
Detection Accuracy	1.0	1.0	0.9	1.0	0.952

Having knowledge of the specifics of a jamming attack can lead to better mitigation, as was demonstrated in the validation phases. The effectiveness of the mitigation method has not been fully demonstrated yet in the user domain; although Antiference does filter jamming signals effectively and shows significantly improved C/N0 values for medium- to high-power jammers, the resulting improvement in the PVT-domain has not been convincingly demonstrated. The main reason for the shortcomings comes from the limitations in the digital domain, as the mitigation/filtering was performed only on the digitized signals, where the degradations of the RFI cannot be fully recovered. In a future evolution, the overall mitigation performance in the user domain can be improved by applying the mitigation directly at the receiver level.

Antiference has been used in combination with two COTS receivers and has been compared to Rec02's internal RFI suppression. As the Rec01's internal RFI mitigation cannot be turned off, Antiference can only be used in combination with it and not compared with it in isolation. The relative performances differ for the different interference types. Antiference performance is typically comparable to that of Rec02, while, in some cases, it is clearly somewhat better or somewhat worse than Rec02. Further optimization is expected to further improve the Antiference performance, but, for example, for FM interference, it already performs remarkably well, as shown in Figure 3 It is, however, hard to distinguish FM and AM interferers; however, due to the identical mitigation strategy, this does not impact performance negatively (see Figure 4).

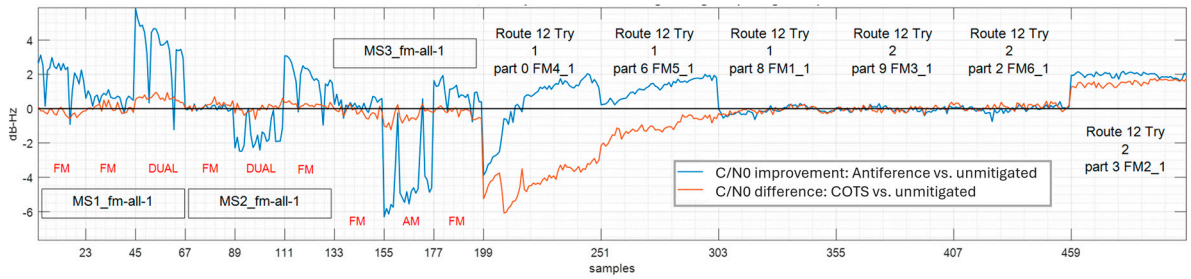

Figure 3. The improvement in observed C/N0 in the presence of FM interference for nine of our recordings (that are depicted head to tail). Antiference shows a better interference suppression than the Rec02 COTS receiver.

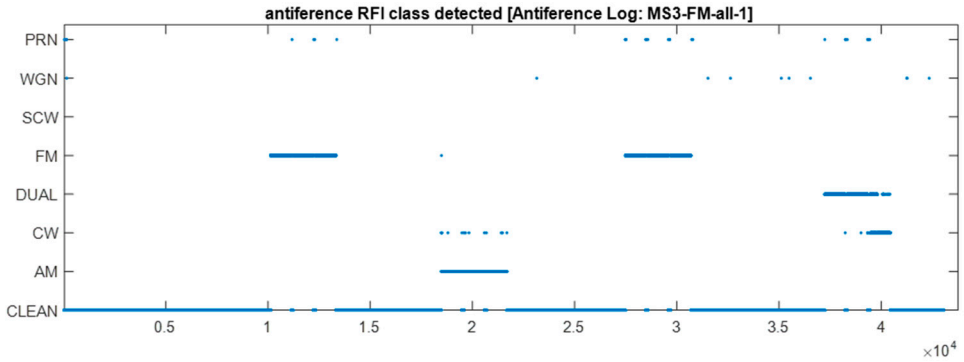

Figure 4. Antiference RFI classification outputs for the first recording from Figure 4. FM interference is hard to distinguish from AM interference and is sometimes mistaken for CW interference. As the mitigation of these types of interference is the same, this does not impact mitigation performance.

In addition, we have looked at the outcomes of the Antiference Logs that provide data on the internal workings of the module, particularly in relation to the false-detection rates. The Antiference algorithms turned out to be highly sensitive, as they effectively detect low-power CW and SCW (chirp) interferences close to 100% of the time. This is not necessarily desirable, as the detection of such low-impact interferences is not needed and comes at the expense of higher-than-needed false-detection rates and lower quality signal due to the removal of some of the signal because of filtering. A simple remedy is the use of a simple majority voting scheme over multiple subsequent RFI classifications. This has been shown to reduce the false detection rates effectively, but at the expense of an increased time-to-detection.

7. Discussion

The Antiference project has been a challenging project. Below, we summarize some of the most important lessons learned.

The detection and classification model has been successful and performs well. However, the system is very sensitive, and picks up very low-power RFI and small perturbations from the environment as well. This has led to a lack of distinguishing between harmless and dangerous RFI. The detection of RFI is made based on small, 10 ms chunks of data. While this solves the problem of detecting pulsed RFI, for the most part, and assures short detection delays, it also leads to some false negatives/positives as 100 detections and classification decisions are made every second. There are many ways this problem can be addressed, such as looking at a window of detections and employing majority voting, or having a secondary model in place that decides on the detection/classification results based on the detection–result streams.

Antiference focused on ML-based methods for the detection and classification model and the fingerprint database. However, the estimation of filter parameters can also be delegated to ML models and might lead to improvements in performance.

There is no standardized way to inform the user of the existence or detected type of RFI. Further developments are needed for meaningful interfacing with GNSS receivers and their operators.

The proper training of ML models requires a massive amount of data. Unfortunately, currently there are not enough free or publicly available recorded datasets that can be used for the training of models. This led to recording campaigns internally, in addition to processing and labelling the data samples manually. However, the amount of data was insufficient for training the models to perform the interference mitigation. In our view, a significantly increased number of GNSS datasets (both clean and containing RFI) is needed for the future widespread development of ML-based RFI detection, classification, and mitigation.

Digital data were used for the training and testing of models. While this was appropriate for detection and classification, the integration of mitigation could not be carried out in this domain. It should be noted that the effective mitigation (that can lead to increased CN/0 in all cases) requires the real-time mitigation of signals in the analogue domain and the inclusion of the effects in the loop. It was suspected that the Antiference system would not be functional in real-time, so the decision was made to keep all processing (barring the COTS receivers) in the digital domain to reduce complexity and keep tests realistic. Until more performant detection and classification models are developed, this will remain a shortcoming of the system.

8. Conclusions

The Antiference project succeeded in the implementation of a prototype, showing the high potential of the application of ML methods to RFI detection and classification. Significant improvements are still needed for the commercialization of such a system. It was convincingly demonstrated that the usage of signal fingerprints and a fingerprint database is feasible and beneficial in documenting the history and streamlining of the performance for subsequent encounters with the same source of RFI.

While the system model developed during the project was a local model, the extended of the general architecture to a cloud-based system can be envisioned, where the detection, classification, and RFI characterization can be carried out remotely on the equipment. The design of very small fingerprints and detection/classification features that can be communicated easily via networks facilitates this approach.

On the other hand, having fingerprints that are explicitly readable by humans facilitates the combination and sharing of databases between equipment. This can result in a better understanding of the growing threats that affect all GNSS receivers in terms of their daily performance and a better cataloguing of the types and frequency of occurrence of different types of RFI.

Author Contributions: Conceptualization, S.A., V.B., A.B. and B.O.; software, V.B. and U.B.; validation, V.B. and U.B.; formal analysis, B.O.; data curation, S.A.; writing—original draft preparation, S.A.; writing—review and editing, A.B., B.O. and S.B.; supervision, S.B.; project administration, S.A.; funding acquisition, S.A., A.B. and B.O. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the European Space Agency, under the program NAVISP Element 1, grant number 4000133535/20/NL/GP.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable.

Data Availability Statement: No datasets are publicly available.

Acknowledgments: The reported work was carried out under ESA NAVISP Element 1, which is devoted to the development of innovative PNT, systems, technologies, algorithms, and techniques.

Conflicts of Interest: The authors declare no conflicts of interest. All involved parties including OHB Digital Solutions, S&T, as well as Integricom declare no conflicts of interest.

References

1. Caparra, G.; Ceccato, P.; Formaggio, F.; Laurenti, N.; Tomasin, S. Low Power Selective Denial of Service Attacks Against GNSS. In Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2018), Miami, FL, USA, 24–28 September 2018; pp. 3028–3041.
2. Curren, J.; Closas, P.; Navarro, M. A look at the Threat of Systematic Jamming of GNSS. *Inside GNSS* **2017**, *12*, 46–53.
3. Lee, G.; Jo, J.; Park, C. Jamming Prediction for Radar Signals Using Machine Learning Methods. *Secur. Commun. Netw.* **2020**, *2020*, 2151570. [CrossRef]
4. Akos, D. Who's Afraid of the Spoofer? GPS/GNSS Spoofing Detection via Automatic Gain Control (AGC). *Navigation* **2012**, *59*, 281–290. [CrossRef]

5. Arjoune, Y.; Salahdine, F.; Islam, S.; Ghribi, E.; Kaabouch, N. A Novel Jamming Attacks Detection Approach Based on Machine Learning for Wireless Communication. In Proceedings of the 2020 International Conference on Information Networking (ICOIN), Barcelona, Spain, 7–10 January 2020; pp. 459–464.
6. Li, W.; Huang, Z.; Lang, R.; Qin, H.; Zhou, K.; Cao, Y. A Real-Time Interference Monitoring Technique for GNSS Based on a Twin Support Vector Machine Method. *Sensors* **2016**, *16*, 329. [CrossRef]
7. Akeret, J.; Chang, C.; Lucchi, A.; Refregier, A. Radio frequency interference mitigation using deep convolutional neural networks. *Astron. Comput.* **2017**, *18*, 35–39. [CrossRef]
8. Carley, E. Using supervised machine learning to automatically detect type II and III solar radio bursts. In Proceedings of the Machine Learning in Heliophysics, Amsterdam, The Netherlands, 16–20 September 2019; p. 11.
9. Ferre, M.; de la Fuente, R.; Lohan, A.; Simona, E. Jammer classification in GNSS bands via machine learning algorithms. *Sensors* **2019**, *19*, 4841. [CrossRef] [PubMed]
10. Semanjski, S.; Semanjski, I.; De Wilde, W.; Muls, A. Use of supervised machine learning for gnss signal spoofing detection with validation on real-world meaconing and spoofing data—Part I. *Sensors* **2020**, *20*, 1171. [CrossRef] [PubMed]
11. Sun, M.; Qin, Y.; Bao, J.; Yu, X. GPS Spoofing Detection Based on Decision Fusion with a K-out-of-N Rule. *Int. J. Netw. Secur.* **2017**, *19*, 670–674.
12. Shafiee, E.; Mosavi, M.; Moazedi, M. Detection of spoofing attack using machine learning based on multi-layer neural network in single-frequency GPS receivers. *J. Navig.* **2018**, *71*, 169–188. [CrossRef]
13. Schmidt, E.; Gatsis, N.; Akopian, D. A GPS spoofing detection and classification correlator-based technique using the LASSO. *IEEE Trans. Aerosp. Electron. Syst.* **2020**, *56*, 4224–4237. [CrossRef]
14. Kokalj-Filipovic, S.; Miller, R.; Morman, J. Autoencoders for training compact deep learning rf classifiers for wireless protocols. In Proceedings of the 2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Cannes, France, 2–5 July 2019; pp. 1–5.
15. O'Shea, T.; Corgan, J.; Clancy, T. Convolutional radio modulation recognition networks. In *Engineering Applications of Neural Networks: Proceedings of the 17th International Conference (EANN 2016), Aberdeen, UK, 2–5 September 2016*; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; Volume 17, pp. 213–226.
16. Shi, Y.; Davaslioglu, K.; Sagduyu, Y.; Headley, W.; Fowler, M.; Green, G. Deep learning for RF signal classification in unknown and dynamic spectrum environments. In Proceedings of the 2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), Newark, NJ, USA, 11–14 November 2019; pp. 1–10.
17. Tang, Z.; Li, S.; Yu, L. Implementation of deep learning-based automatic modulation classifier on FPGA SDR platform. *Electronics* **2018**, *7*, 122. [CrossRef]
18. Shah, H.; Koo, I. Reliable machine learning based spectrum sensing in cognitive radio networks. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 5906097. [CrossRef]
19. de Vriez, C.; Simic, L.; Mahonen, P. The importance of being earnest: Performance of modulation classification for real RF signals. In Proceedings of the 2018 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), Seoul, Republic of Korea, 22–25 October 2018; pp. 1–5.
20. Dovis, F. *GNSS Interference Threats and Countermeasures*; Artech House: Norwood, MA, USA, 2015.
21. Regalia, P. An improved lattice-based adaptive IIR notch filter. *IEEE Trans. Signal Process.* **1991**, *39*, 2124–2128. [CrossRef] [PubMed]
22. Regalia, P. A complex adaptive notch filter. *IEEE Signal Process. Lett.* **2010**, *17*, 937–940. [CrossRef]
23. Chen, Y.; Chien, Y.; Tsao, H. Chirp-like jamming mitigation for GPS receivers using wavelet-packet-transform-assisted adaptive filters. In Proceedings of the 2016 International Computer Symposium (ICS), Chiayi, Taiwan, 15–17 December 2016; pp. 458–461.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
25. Keras Applications. Available online: https://keras.io/api/applications/ (accessed on 5 April 2023).
26. KerasTuner API. Available online: https://keras.io/api/keras_tuner/ (accessed on 5 April 2023).
27. Keras BayesianOptimization Tuner. Available online: https://keras.io/api/keras_tuner/tuners/bayesian/ (accessed on 5 April 2023).