Deep Neural Network approach to detect GNSS Spoofing Attacks

Parisa Borhani-Darian, *Electrical and Computer Engineering Dept., Northeastern University*, Boston, MA, USA.
Haoqing Li, *Electrical and Computer Engineering Dept., Northeastern University*, Boston, MA, USA.
Peng Wu, *Electrical and Computer Engineering Dept., Northeastern University*, Boston, MA, USA.
Pau Closas, *Electrical and Computer Engineering Dept., Northeastern University*, Boston, MA, USA.

BIOGRAPHIES

Parisa Borhani-Darian is received her BS. and MS. degrees in Computer Engineering. She currently is a PhD candidate in the Department of Electrical and Computer Engineering at Northeastern University, Boston, MA. Her current research is working on safety in CAVs in the area of GNSS positioning error and GNSS spoofing attack also she is working on GNSS signal acquisition.

Haoqing Li got his BS degree in Electrical Engineering from Wuhan University, China and MS degree in Electrical and Computer Engineering at Northeastern University, Boston, MA. Currently, he is completing his PhD studies in Electrical and Computer Engineering at Northeastern University, Boston, MA. His research interests include GNSS signal processing, anti-jamming technology and robust statistics.

Peng Wu received his BS degree in Physics from Tianjin University of Technology, China and MS degree in Electrical Engineering from Northeastern University, Boston, MA. He is currently a PhD candidate in the Department of Electrical and Computer Engineering at Northeastern University. His research interests include machine learning with applications to indoor positioning and tracking.

Pau Closas is Assistant Professor at Northeastern University, Boston, MA. He received the MS and PhD degrees in Electrical Engineering from UPC in 2003 and 2009. He also holds a MS in Advanced Mathematics from UPC, 2014. His primary areas of interest include statistical signal processing, robust stochastic filtering, and machine learning, with applications to positioning systems and wireless communications. He is the recipient of the 2014 EURASIP Best PhD Thesis Award, the 9th Duran Farell Award, the 2016 ION Early Achievements Award, and a 2019 NSF CAREER Award.

ABSTRACT

This article discusses the use of deep learning schemes for spoofing detection. Particularly, the characteristics of the so-called Cross Ambiguity Function (CAF) in the presence and absence of spoofing signals are exploited to train a set of data-driven models providing a probabilistic classification. The method operates on a per-satellite basis. The results show that complex neural networks are effectively able to capture the nature of spoofing attacks. Particularly, a Multi-Layer Perceptron (MLP) and two classes of Convolution Neural Networks (CNNs) are considered in this work, validated over simulated data.

INTRODUCTION

Location-based services, alongside with the modern applications on Intelligent Transportation Systems require reliable, continuous and precise navigation, positioning and timing information for their successful operation and implantation in the market. Global Navigation Satellite Systems (GNSS) constitute the backbone and main information supplier for Positioning, Navigation and Timing (PNT) data [2, 6, 7].

GNSS receivers are very sensitive and vulnerable to deliberate interference, which opens the door for attackers who want to compromise a GNSS-based system or infrastructure causing serious impacts. Due to the lack of inherent

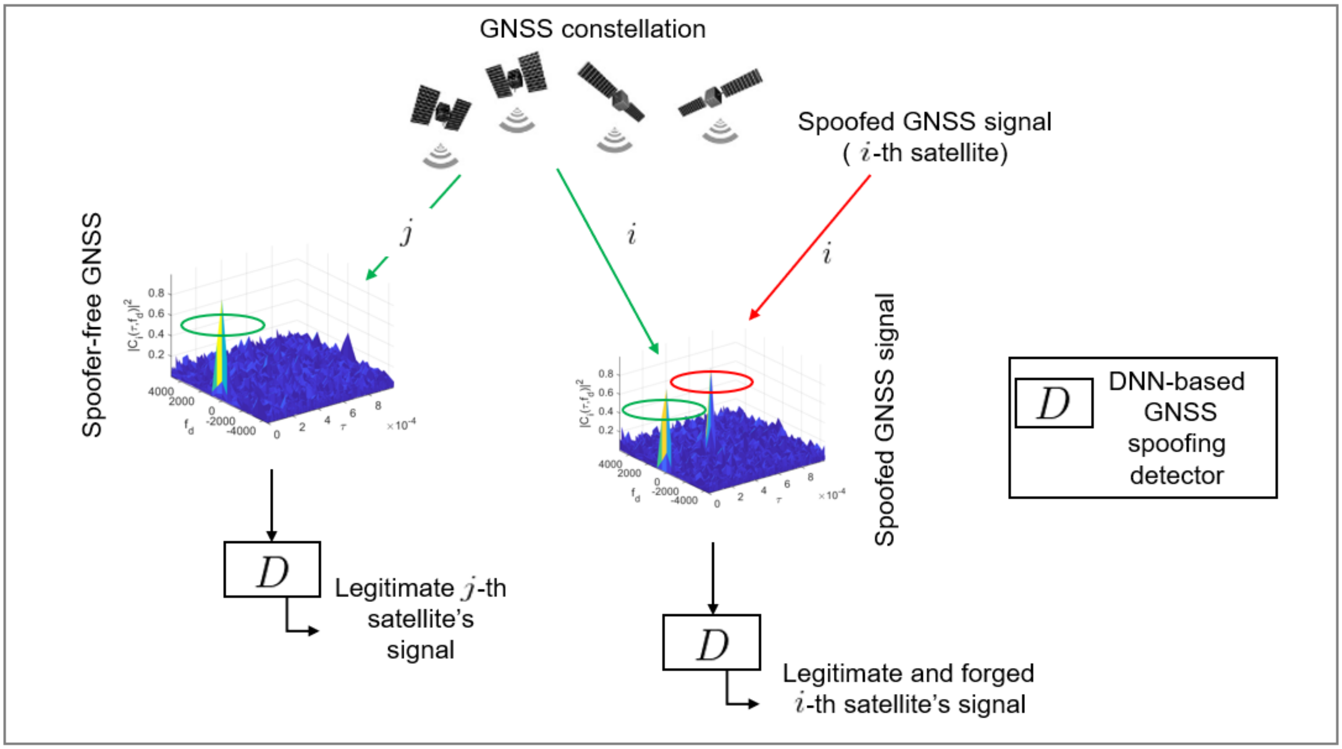
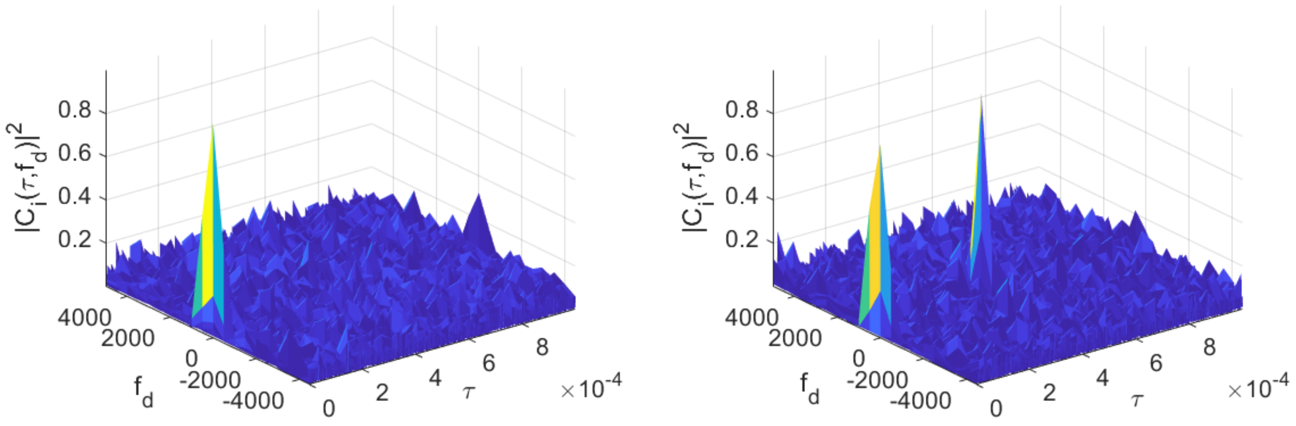


Figure 1: Deep learning signal detection process in the presence of spoofer

security design in GNSS systems, many applications could be potentially at risk as reported in numerous articles. Deliberately attacks on GNSS receivers might act at two different categories: physical attacks on the receiver (non-signal attacks) or attacks at the GNSS signal-in-space (SIS) level (signal attacks) [8]. This paper is working on the second category (signal attacks), which are intentional attacks on GNSS signals, which has three different forms: jamming, meaconing, and spoofing. The focus of this work is on spoofing, which is the transmission of forged GNSS-like signals, with the purpose to produce a false position at the victim's receiver without disrupting GNSS operations, effectively taking control of the receiver. Notice that jamming attacks aim at denying GNSS positioning service, an opposite goal to spoofing interference. In [20] to obtain a higher detection probability of the GPS spoofing, a general identification scheme with decision fusion is used. The singular values of the wavelet transform coefficients of both spoofing and genuine signal considered as feature vectors, that are input into three classifiers, which are the support vector machines (SVM), the probabilistic neural networks (PNN) and the decision tree (DT), respectively, for GPS spoofing identification. The results of the three classifiers are fused with a K-out-of-N decision rule, and the final classification result has a higher probability detection and lower false alarm. This [17] presents a method to detect a GPS spoofing based on Multi-Layer NN whose inputs are indices of features to performed spoofing detection by exploiting conventional machine learning algorithms such as K-Nearest Neighbourhood (KNN) and naive Bayesian classifier. In [10], two methods of predicting the appropriate jamming technique for a received threat signal using deep learning is investigated: using a deep neural network on feature values extracted manually from the PDW list and using long short-term memory (LSTM) which takes the PDW list as input. [14] work proposed to treat the jammer classification problem in the Global Navigation Satellite System bands as a black-and-white image classification problem, based on time-frequency analysis and image mapping of a jammed signal. It also proposed to apply machine learning approaches to sort the received signal into six classes, namely five classes when the jammer is present with different jammer types and one class where the jammer is absent.

The goal of this work is to detect the GNSS signal spoofing attack. In particular, the paper proposes to use a deep neural network (DNN) to carry out the detection (or classification) task. Deep learning technique has been recently evolving in various fields, including autonomous vehicle navigation. This paper aims at exploring the capabilities of deep learning models in complementing the standard design of GNSS receivers, in particular, it focuses on the



(a) legitimate signal, $\mathcal{H}_0$

(b) legitimate and spoofing signals, $\mathcal{H}_1$

Figure 2: CAF evaluation at the Delay/Doppler grid

signal detection process in presence of spoofer which to the best of our knowledge has not been explored in the past, which is shown in Fig.1. Two different hypotheses considered, the null hypothesis ($\mathcal{H}_0$), that the legitimate signal and noise are present, but there is no spoofing signal and the alternative hypothesis ($\mathcal{H}_1$), that both the legitimate signal, spoofed signal, and noise are present in the dataset.

To this end, three different neural networks designed and evaluated the two hypotheses detection process such as Complex-CNN, Simple-CNN, and MLP. The networks are trained with two different optimizers for various C/N_0 between 33 to 45 dB-Hz at a different delay and Doppler frequency with 1 ms coherent and 10 non-coherent integration configuration. The results show that the Complex-CNN outperforms the Simple-CNN and MLP in training accuracy and also in the detection process. The Simple-CNN and MLP have a very low probability detection in lower C/N_0 in comparison with Complex-CNN, which can be extracted from histogram figures since the low C/N_0 completely overlapped it shows that distinguishing these two hypotheses from each other almost is impossible.

The paper is organized as follows. GNSS SPOOFER DETECTION section reviews the GNSS Spoofed signal model and the standard model of the Spoofed signal. The data-driven approach introduces and connects to the problem of GNSS Spoofed signal in DEEP NEURAL NETWORKS MODEL AND PROBLEM STATEMENT. SIMULATION ENVIRONMENT AND RESULTS SECTION explain the details of the training of a neural network and discuss the results. The paper is concluded in the final remarks in CONCLUSION.

GNSS SPOOFER DETECTION

To understand the impact of a spoofing attack - and its potential countermeasures -, we first recall that the first stage in the operation of a Global Navigation Satellite System (GNSS) receiver is acquisition. This process results in deciding whether the satellite signal is present or absent in the received signal, as well as provides rough estimates of the parameters, such as code delay and Doppler shift of the signal transmitted by the satellite. All GNSS receivers [4, 9, 13, 21] implement such an acquisition process by evaluating the so-called Cross Ambiguity Function (CAF), usually in discrete-time.

In order to detect the presence of visible satellite signals, a receiver performs a two-dimensional search. The acquisition process involves calculating the CAF value for a given satellite by correlating the received signal and local pseudo-random code for every possible delay/Doppler pair in the search space. When the signal is present, the CAF exhibits a large peak for a specific delay/Doppler cell, which is then passed on to the tracking loops for fine estimation and tracking. Fig. 2 shows the exemplary representation of this process.

Received GNSS signal is considered as:

$$X(t) = \sum_{i=1}^{N} \alpha_i b_i(t - \tau_i(t)) c_i(t - \tau_i(t)) \exp(j(\omega_{IF} t - \omega_c \tau_i(t) - \phi_i(t)))$$

where N is the number of spreading code, $\alpha_i(t)$ is the carrier amplitude of the i-th signal, b_i is the i-th signal's data bit stream, $c_i(t)$ is its spreading code, $\tau_i(t)$ is the i-th signal's code phase, ω_c is the carrier frequency, ω_{IF} the intermediate frequency, and $\phi_i(t)$ is the i-th carrier phase. The spoofer sends a set of false signals that are similar to true signal, except for those parameters that would eventually cause a different position estimate at the receiver unless properly detected.

A receive GNSS spoofed signal is:

$$X_s(t) = \sum_{i=1}^{N_s} \alpha_{s,i} b_i(t - \tau_{s,i}(t)) c_i(t - \tau_{s,i}(t)) \exp(j(\omega_{IF} t - \omega_c \tau_{s,i}(t) - \phi_{s,i}(t)))$$

where N_s, denotes the number of spoofed signals. In order to deceive the receiver, each spoofed signal must have the same spreading code $c_i(t)$ as the corresponding true signal and broadcast its best estimate of the same data bit stream b_i, the spoofed amplitude $\alpha_{s,i}(t)$, code phases $\tau_{s,i}(t)$, and carrier phases $\phi_{s,i}(t)$ for $i = 1, \ldots, N$ are different from those of the true signal [16].

The total signal at the victim receiver antenna during a spoofing attack is:

$$Y(t) = X(t) + X_s(t) + \eta(t) \tag{1}$$

where $\eta(t)$ is the received noise, typically modeled as zero-mean, additive, white, and Gaussian.

Therefore, two hypotheses are tested:

1. The null hypothesis ($\mathcal{H}_0$), that the legitimate signal and noise are present, but there is no spoofing signal.

$$\mathcal{H}_0 : Y(t) = X(t) + \eta(t)$$

2. The alternative hypothesis ($\mathcal{H}_1$), that both the legitimate signal, spoofed signal, and noise are present in the dataset;

$$\mathcal{H}_1 : Y(t) = X(t) + X_s(t) + \eta(t)$$

In this context, we propose to treat the spoofing detection problem as a hypothesis testing problem on the CAF's delay/Doppler map for each satellite independently, for instance the i-th satellite. After downconversion and sampling (at a rate $f_s = 1/\mathrm{T}_s$), the CAF can be computed as the correlation of the digital signal and the known local code for the i-th satellite. At a given delay/Doppler pair:

$$\mathcal{C}_i(\tau, f_d) = \frac{1}{N} \sum_{n=0}^{N-1} y[n] \underbrace{c_i(n\mathrm{T}_s - \tau) \exp\{-j2\pi f_{d,i} n \mathrm{T}_s\}}_{\text{Local replica}} , \tag{2}$$

which can be expressed more compactly in vector notation after gathering N samples from the samples and the local code as $\mathbf{y}, \mathbf{c}_i \in \mathbb{C}^{1 \times N}$ as

$$\mathcal{C}_i(\tau, f_d) = \frac{\mathbf{y}\mathbf{c}_i^H}{N} . \tag{3}$$

The effect of a spoofing signal on the CAF is well-known and shown in Fig. 2 for clarity, showing an arbitrary CAF under both hypotheses. This work proposes to train a DNN, data-driven model to learn to classify between spoofed or clean signal receptions.

Three types of error probabilities characterize the performance of the detector: detection (the probability of correctly detecting signal+spoofer when there is signal+spoofer), false-alarm (the probability of wrongly detecting satellite is spoofed when there is only signal), and miss-detection (the probability of mistakenly deciding for the null hypothesis when the satellite is spoofed). The two first probabilities are used in order to obtain an important figure of merit in acquisition performance: the Receiver Operating Characteristics (ROC), a plot of the probability of detection as a function of the probability of false alarm [4, 12, 22]. We will use ROC curves in evaluating the performance of the proposed detection method.

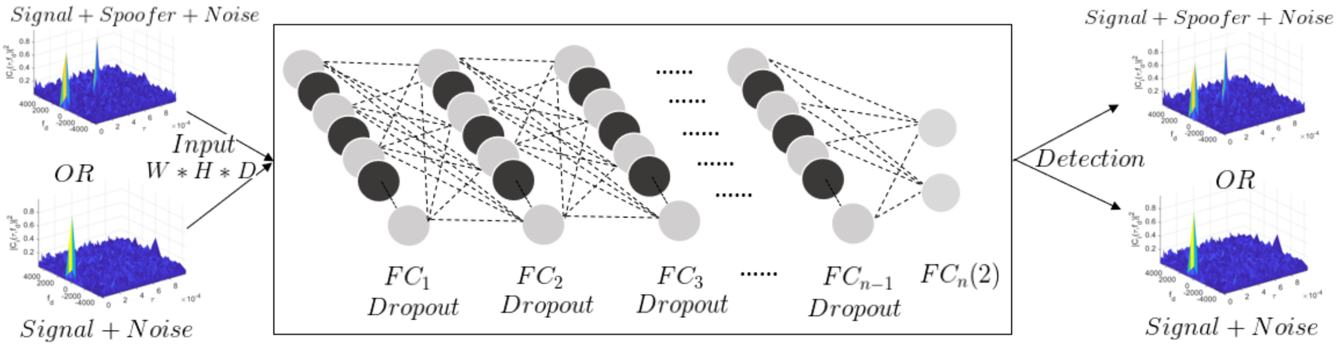


Figure 3: Detection scheme of GNSS acquisition by using the Fully Connected Structure of Deep Learning approach.

DEEP NEURAL NETWORKS MODEL AND PROBLEM STATEMENT

Neural networks (NN) are models composed of neurons, which are information processing units, for complex data processing. A NN typically contains an input layer, one or more hidden layers, and an output layer, as well as pre-defined activation functions that connect adjacent layers. Each layer has a specific weight, which is usually determined with backpropagation during a training process that involves large amounts of data with known labels [15, 23]. One of the most important part in NN is how to design and choose a network to get the high accuracy with less network complexity. some of effective aspects are: number of layers, number of neurons, and type of optimizer. In the simulation and result section, the accuracy of different types of networks is discussed more. In this work, in order to detect the spoofed signal, the performance of two different structure of neural networks are evaluated and compared together: a Multi-Layer Perceptron (MLP) and a Convolution Neural Network (CNN).

MLP Network

The first neural network structure that will be used in this research is the Multi-Layer Perceptron (MLP), which is referred to as a traditional neural network. This type of network is comprised of one or more layers of neurons (which consist of a row of neurons). Fig.3 shows an exemplary representation of this type of network. The first layer is called the input layer, which is fed with the training dataset for learning the parameters of the (potentially several) hidden layers that are not directly exposed to the input. During training, the number of nodes in the hidden layer are randomly ignored or "dropped out", which are shown in black in Fig.3. The number of neurons in this last layer depends on the number of the classes that one wants to classify since those provide their probabilities.

CNN Network

The second artificial neural network structure considered here is the so-called Convolutional Neural Network (CNN), which is one of the most popular models for deep learning in the context of learning class labels from image datasets. A CNN can have tens or hundreds of layers, where each of these layers learns to identify different features of an image [19]. At each layer filters are applied to each training image and the output of each convolution image is used as an input to the next layer. Fig.4 illustrates a CNN structure. During training, the input size of the CNN is fixed, the input is going through a stack of convolutional layers with the same or different filter sizes. In each convolution layer, the filter sweeps the input image from left to right and up to down by using stride with 2 pixels size, which is the number of pixels that each time the filter shifts. In the end, the convolution layers are followed by Fully Connected (FC) layers and a final softmax layer, which is used for classification purposes [19].

Probabilistic learning

The aforementioned DNN models have the task of classifying CAF maps into spoofed/cleaned cases. We aim at doing that on a probabilistic sense, thus providing probabilities for both hypotheses. The inputs to the NNs is the CAF evaluated at the delay/Doppler grid, which can be considered as an *image*. Such images (refer to Fig. 2 for an exemplary situation) has certain characteristics that can be used to determine whether the signal is present or not: *i*) in the absence of spoofing signal, the image should exhibit a single peak corresponding to the legitimate

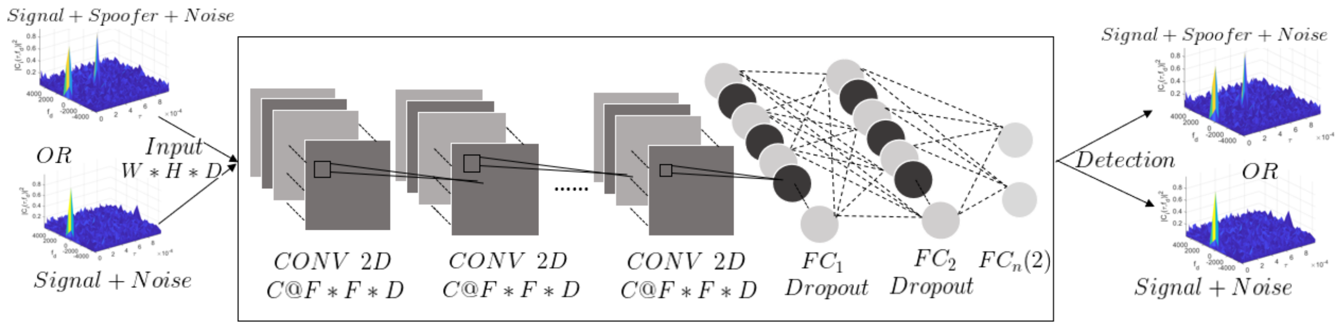


Figure 4: Detection scheme of GNSS acquisition by using the Convolutional Neural Network Structure of Deep Learning approach.

satellite signal (if received with enough power); and ii) in the presence of a spoofing signal, the CAF *image* should be composed of two peaks and exponentially distributed noise in the remaining cells. This is used to train a NN model to classify between $\mathcal{H}_0$ and $\mathcal{H}_1$, the hypotheses described earlier.

The proposed methodology works on a per-satellite basis. Recall that the input data fed to the NNs is the corresponding CAF image for the satellite, which we denote with $\mathbf{Z}_i$ in the sequel. That is, the $\{m, n\}$ element of the input matrix is defined as

$$[\mathbf{Z}_i]_{m,n} = \left|\mathcal{C}_i\left([\boldsymbol{\tau}]_m, [\mathbf{f}_d]_n\right)\right|^2 , \tag{4}$$

where $\boldsymbol{\tau}$ and $\mathbf{f}_d$ are vectors containing the tested delay and Doppler-shifts, respectively. We use the convention that $[\mathbf{a}]_m$ represents the m-th element in vector $\mathbf{a}$ and that $[\mathbf{A}]_{m,n}$ provides a shortcut for the element of $\mathbf{A}$ in the m-th row and n-th column.

From a Bayesian perspective, all the information resides in the *a posteriori* distribution of each hypothesis once data is observed. An optimal (Bayesian) test between $\mathcal{H}_0$ and $\mathcal{H}_1$ is given by the ratio

$$\frac{p(\mathcal{H}_1|\mathbf{Z}_i)}{p(\mathcal{H}_0|\mathbf{Z}_i)} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} 1 , \tag{5}$$

in which case we basically favor the model with largest a posteriori probability. This can be further expanded in terms of the likelihood and a priori distributions as

$$\frac{p(\mathbf{Z}_i|\mathcal{H}_1)}{p(\mathbf{Z}_i|\mathcal{H}_0)} \frac{\mathbb{P}(\mathcal{H}_1)}{\mathbb{P}(\mathcal{H}_0)} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} 1 , \tag{6}$$

where we readily identify that $\mathbb{P}(\mathcal{H}_i)$ denotes the a priori probability of the i-th hypothesis. In the absence of better priors, we may assume equally likely hypotheses $\mathbb{P}(\mathcal{H}_0) = \mathbb{P}(\mathcal{H}_1) = 1/2$. Otherwise, we might incorporate that information in the hypothesis test, resulting on the adjustment of a threshold γ. The resulting test statistic is such that

$$\mathcal{T}(\mathbf{Z}_i) \triangleq \frac{p(\mathbf{Z}_i|\mathcal{H}_1)}{p(\mathbf{Z}_i|\mathcal{H}_0)} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \gamma , \tag{7}$$

which would act as our spoofer detection algorithm, where threshold γ is a tuning parameter. Since the test statistic is a ratio of probabilities, we have that $0 < \mathcal{T}(\mathbf{Z}_i) < \infty$. Similarly as in [3], the DNNs provide estimated probabilities for each of the hypotheses in (7). Therefore, the input data would be $\mathbf{Z}_i$ and the output of the DNN would be the estimated probabilities in the dataset $\mathbf{y}$ used to build $\mathbf{Z}_i$.

SIMULATION ENVIRONMENT AND RESULTS

To accomplish the objective, we train DNNs using synthetically generated data. Particularly, we use a dataset consisting of 10000 snapshots of I&Q samples with different Carrier-to-Noise-density ratio (C/N_0) varying between 33 to 45 dB-Hz, as well as randomly generated delays between 0 to 1 ms and Doppler shifts between -4000 to 4000 Hz. These samples are then processed to compute the CAF over a Doppler-delay grid. An analogy to images can

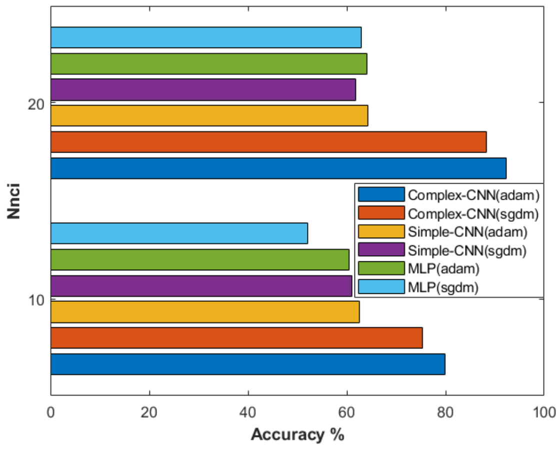


Figure 5: Neural networks training accuracy for $C/N_0 = [33 - 45]$ dB-Hz

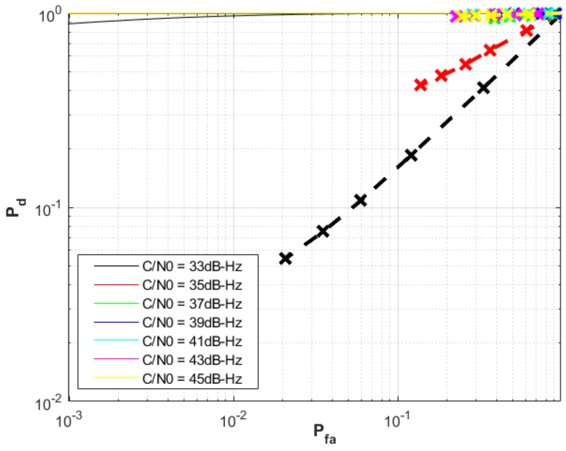
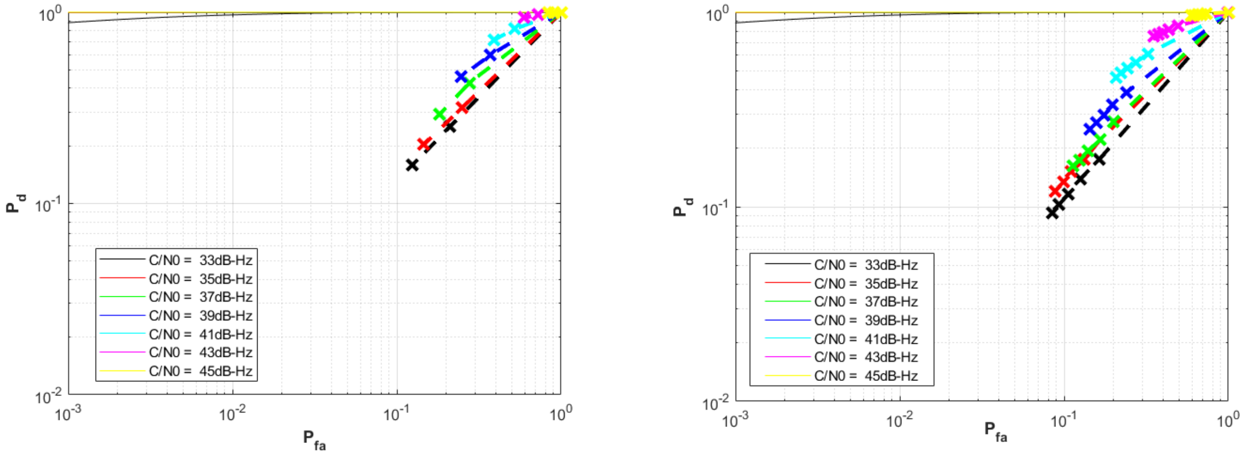


Figure 6: ROC Curve Complex-CNN (VGG16) for $C/N_0 = [33 - 45]$ dB-Hz



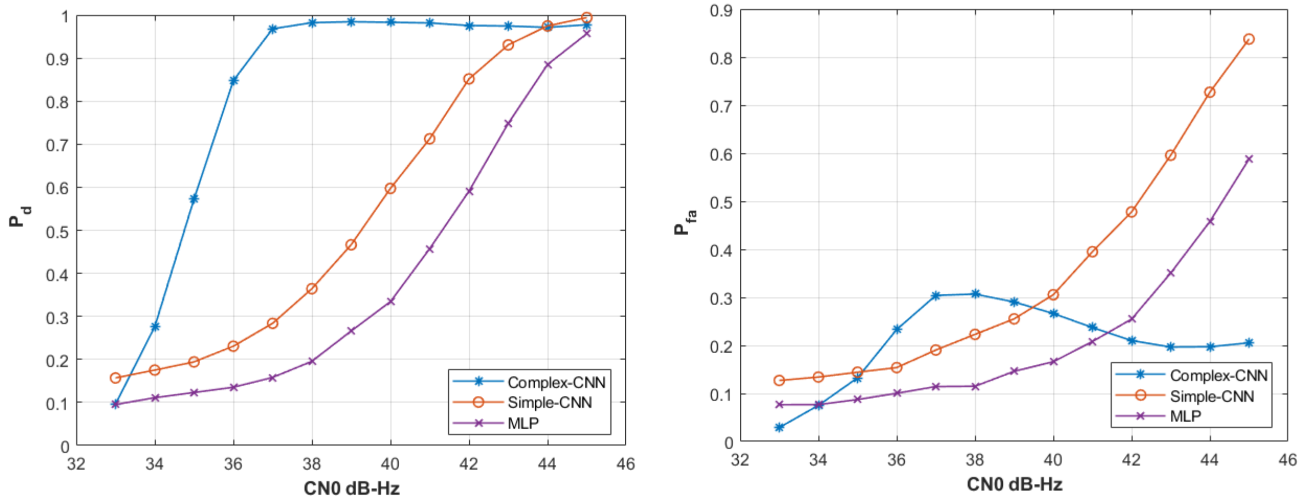
(a) Simple-CNN for $C/N_0 = [33 - 45]$ dB-Hz

(b) MLP for $C/N_0 = [33 - 45]$ dB-Hz

Figure 7: ROC curves for Simple-CNN and MLP using CAF generated with 1 ms coherent integration and 10 non-coherent integration

be made for these CAFs where each Doppler/delay cell is a pixel whose value is that of the CAF. For instance, if 20 Doppler bins are considered to acquire a GPS L1 C/A signal, those images would be 20 × 1023 dimensional. These images are fed to the input layer of the DNN, whose output would be the classification into a present/absent of the spoofed satellite signal. In a supervised training scheme, these input/output pairs are provided by labeling and using the aforementioned synthetic data generation. In the experiments, the ReLU activation function and two different optimize are used to train the network Stochastic Gradient Descent with Momentum (SGDM) and Adam optimizer. In this work, two different NN structures will be considered: Multilayer perceptron (MLP), and Convolution Neural Network (CNN).

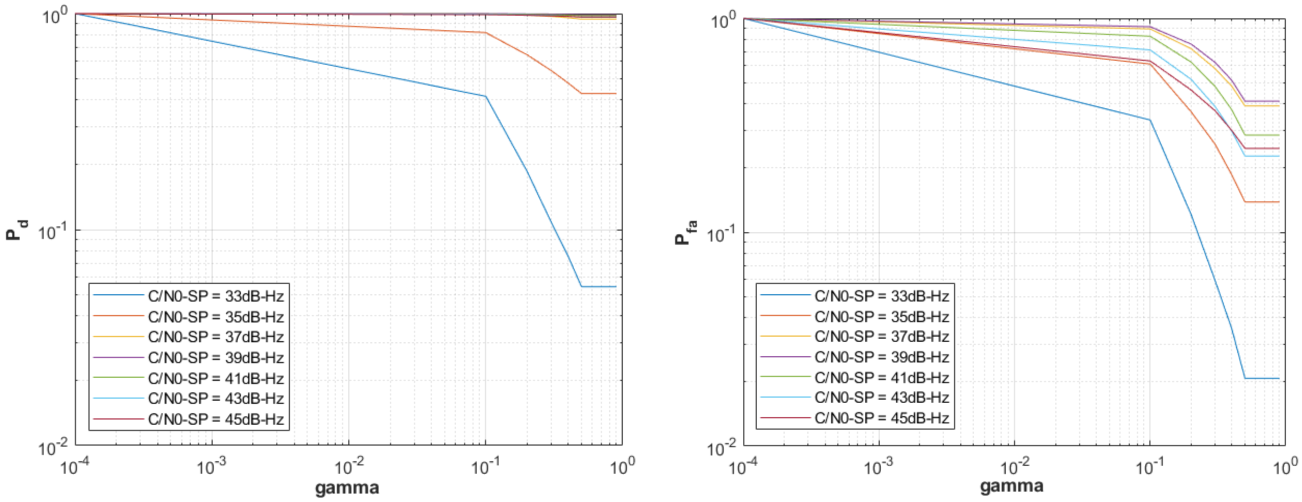
The model implemented with MLP structures considered different number of fully connected layers. Each fully connected layer followed up with rectified linear unit (ReLU) activation function to allows for faster and more effective training by mapping negative value to zero [18]. After each layer a 1/2 dropout probability was considered. The last fully connected layer contained two neurons, used in predicting the class for which the input image belongs to. After defining the network structure, the training options were specified. The network trained with two different optimizer Stochastic Gradient Descent with Momentum (SGDM) and Adam optimizer with an initial learning rate of 0.001.



(a) Probability of detection for different C/N_0

(b) Probability of false alarm for different C/N_0

Figure 8: P_d and P_{fa} under 1 ms coherent and 10 non-coherent integrations for all three networks models.



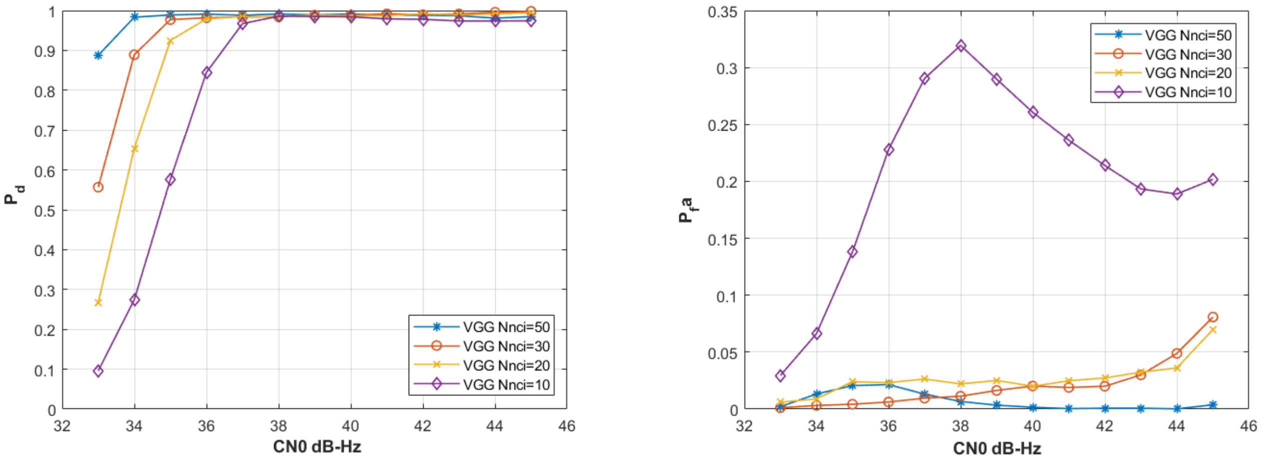
(a) Probability of detection with different threshold

(b) Probability of false alarm with different threshold

Figure 9: $P_d(\gamma)$ and $P_{fa}(\gamma)$ under 1 ms coherent and 10 non-coherent integration for Complex-CNN models.

The maximum number of epochs, which is a full training cycle on the entire training dataset, was set to 30 and at every epoch the data was shuffled. After 20 epochs the learning rate dropped by factor of 0.1. the training progress shows the mini-batch loss and accuracy and the validation loss and accuracy. The loss is the cross-entropy loss and the accuracy was defined as the percentage of inputs that the network classified correctly.

In this work, two different CNN structures are used. First, the simple CNN structure is used and evaluated to detect the Spoofed signal, which contains 3 convolution layers and 3 fully connected layer. Each convolution layer was followed by a batch normalization layer and a ReLU activation function. The batch normalization layers, normalizing the activation and gradients propagation through a network and using it between the convolution layer and ReLU layers to speed up network training [1]. Each fully connected layer follows up with the ReLU activation function and a dropout layer with a probability of 0.5. The last fully-connected layer contains two neurons to predict each image belongs to which class since two types of classes are exist in this work. The result of this network was not promising since as it was said in [3] it is expected that CNN outperforms the MLP to detect the Spoofed signal. Therefore, the more complex CNN structure is decided to use and evaluated to detect the Spoofed signal. The complex CNN



(a) Probability of detection for different $\mathrm{C/N_0}$

(b) Probability of false alarm for different $\mathrm{C/N_0}$

Figure 10: $P_d(\gamma)$ and $P_{fa}(\gamma)$ under 1 ms coherent and 10,20,30 and 50 non-coherent integration for Complex-CNN models.

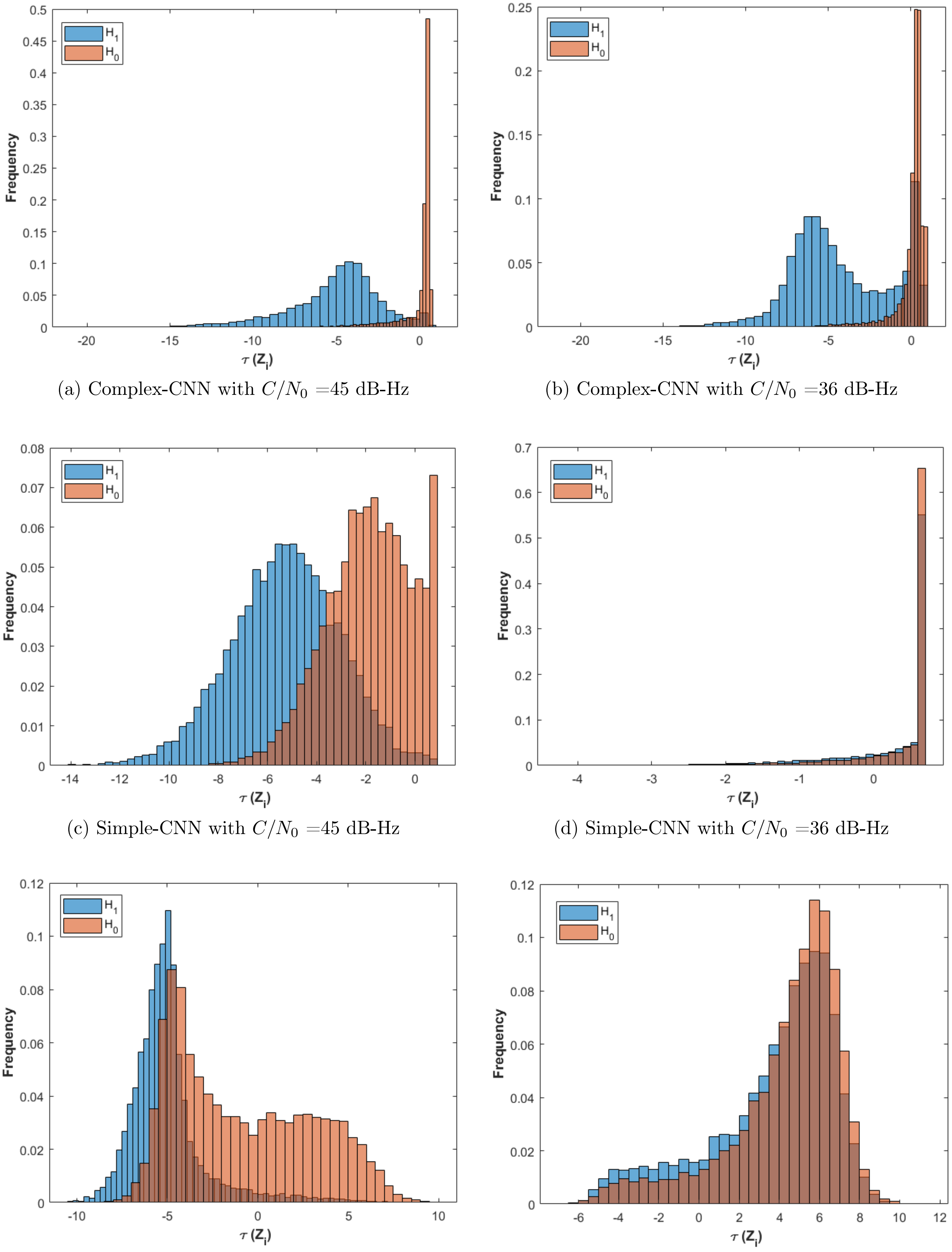
structure which is used in this research is very similar to the VGG 16 structure training [11]. The network has 13 convolution layers that each layer was followed by a batch normalization layer and a ReLU activation function. Also, it contains 3 fully connected layers that each layer follows up with the ReLU activation function and a dropout layer with a probability of 0.5. The last fully-connected layer contains two neurons to predict each image belongs to which class since two type of classes are exist in this work. In the end, after defining the network structure, the training options were specified, which were the same as for the MLP training options. Fig. 5 shows the training accuracy of these three networks with Adam and SGDM optimizer for C/N_0 between 33 to 45 dB-Hz with 2 different number of non-coherent integration (Nnci) values. The result shows that the Complex-CNN (VGG16) network outperforms Simple-CNN and MLP. Moreover, in all networks the Adam optimizer has better accuracy than the SGDM optimizer.

The detection process determines the presence or absence of the Spoofer and the output is the random variable which is called a decision variable. If the Spoofer is present, The probability that the decision variable passes a threshold β is called the detection probability and if the Spoofer is absent it called false alarm probability. Then the plot of detection probability (P_d) versus the false alarm probability (P_{fa}) is called the Receiver Operating Characteristic (ROC). Fig. 6 and 7 show the Receiver Operating Characteristic (ROC) curve result that they are trained and tested under the same integration configuration for Complex-CNN, Simple-CNN and MLP respectively.

From Fig. 6 it can be observed that for low C/N_0 values the performance of NNs is not attaining the theoretical ROC curves. However, as C/N_0 increases, such an approach can reach theoretical limits. An explanation could be that for low C/N_0 the various NNs cannot extract the relevant features from the input images. When C/N_0 increased, either because of an actual power increase or longer integration times, the NN is able to perform classification on whether the Spoofer present or not. In comparing the ROC curve result of Fig. 6 with Fig. 7 it is shown that Complex-CNN outperforms the Simple-CNN and MLP network structures. The Simple-CNN and MLP are not able to reach the theoretical even in higher C/N_0. Indeed, these results were predictable from Fig. 5 since the neural network can be trained much better with Complex-CNN, which is 80%, in comparison with Simple-CNN and MLP, which are around 60%.

As it was discussed in [3] for simple scenario and dataset the MLP and Simple-CNN are working similarly. However, for a more complex scenario, CNN outperforms the MLP because in CNN the input pass through a set of convolution filters, each of these filters activates certain features from the images and creates an output that causes to improve the detection. In order to, increase this feature designing more Complex-CNN would be helpful which is discussed in this work.

Fig. 8 shows the probability of detection and probability of false alarm of different C/N_0 for all 3 networks. As was discussed, the Complex-CNN has the best probability of the detection for all different C/N_0, which is close



(a) Complex-CNN with C/N_0 =45 dB-Hz

(b) Complex-CNN with C/N_0 =36 dB-Hz

(c) Simple-CNN with C/N_0 =45 dB-Hz

(d) Simple-CNN with C/N_0 =36 dB-Hz

(e) MLP with C/N_0 =45 dB-Hz

(f) MLP with C/N_0 =36 dB-Hz

Figure 11: Histogram of $\mathcal{T}(\mathbf{Z}_i)$ under $\mathcal{H}_0$ and $\mathcal{H}_1$ hypotheses for each NNs and two different C/N_0 values.

to 90% for $C/N_0 = 36$ dB-Hz and close to 100% for $C/N_0 = 37$ dB-Hz and greater. Moreover, Simple-CNN and MLP have a worse probability of detection than Complex-CNN even at higher C/N_0. However, in this situation Simple-CNN outperforms MLP.

In Fig. 9, detection and false alarm probabilities are plotted against the classification threshold (i.e., $P_d(\gamma)$ and $P_{fa}(\gamma)$), for different values of C/N_0 under 10 non-coherent integration for complex-CNN network models. It can be taken from these figures the probability of detection and probability of false alarm for all different C/N_0 converged when the (γ) is equal to 0.5.

The result of probability of detection and probability of false alarm at the convergence point $(\gamma = 0.5)$ for different C/N_0 under 1 ms coherent and 10, 20, 30 and 50 non-coherent integration for Complex-CNN models is depicted in Fig. 10(a), 10(b) respectively. It can be taken from the figures by increasing the number of non-coherent integration the probability of detection increased and the probability of false alarm decreased significantly. As it shows in fig. 10(a), the detection probability for C/N_0 38 and above for all number of non-coherent integration are the same and close to 100%, but increasing the number of non-coherent integration affected the result of lower C/N_0 significantly. For example, for C/N_0 33 when the number of non-coherent integration is equal to 10 the probability of detection is almost 10%, however, by increasing it to 50 the probability of detection reach the 90%. A similar improvement happened for the probability of false alarm in Fig. 10(b) that show when the number of non-coherent integration is equal to 10 the probability of false alarm is almost 20% even for highest C/N_0, however, by increasing it to 50 the probability of false alarm decrease to the 0 for almost all C/N_0.

The impact of low C/N_0 on ROC performance is further explained. The histograms under $\mathcal{H}_0$ and $\mathcal{H}_1$ are shown in Fig. 11 for all 3 networks under two different C/N_0. Ideally, one would like to have as few histogram overlapping as possible, such that they can be easily distinguished. Fig. 11(d) & 11(f) shows the Simple-CNN and MLP networks for C/N_0= 36 respectively, as the result got from Fig. 8(a) for lower C/N_0 of these two networks P_d is too low, that the reason is obvious in histogram plot since two hypotheses completely overlapped, which make the detection almost impossible.

CONCLUSIONS

This article presents a DNN approach to detect spoofing attacks. Spoofing attacks are, in general, difficult to model and counteract [5]. In those situations, data-driven schemes become useful if enough training data is available. This article explores this approach using the CAF delay/Doppler map as an input to a DNN for classification purposes. Particularly, several neural network models are trained with and their performance compared in terms of detection and false alarm probabilities. Results show promising performances, particularly with more complex NNs, able to capture the nature of spoofing attacks and their impact on CAF maps. Future works will revolve around improving those results in terms of lower false alarm probability, while keeping detection probabilities large.

ACKNOWLEDGEMENTS

This work was partially supported by the National Science Foundation under Awards CNS-1815349 and ECCS-1845833, as well as a NU's FY21 TIER 1 Award.

REFERENCES

[1] Create simple deep learning network for classification. https://www.mathworks.com/help/deeplearning/examples/create-simple-deep-learning-network-for-classification.html.

[2] M. G. Amin, P. Closas, A. Broumandan, and J. L. Volakis. Vulnerabilities, threats, and authentication in satellite-based navigation systems [scanning the issue]. *Proceedings of the IEEE*, 104(6):1169–1173, 2016.

[3] P. Borhani-Darian and P. Closas. Deep Neural Network Approach to GNSS Signal Acquisition. In *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pages 1214–1223. IEEE, 2020.

[4] D. Borio. A statistical theory for GNSS signal acquisition. *PhD Dissertation Polytecnico di Torino*, 2008.

[5] P. Closas, J. Arribas, and C. Fernández-Prades. Spoofing detection by a reduced acquisition process. In *Proceedings of the Precise Time and Time Interval Systems and Applications Meeting (ION PTTI 2016)*, 2016.

[6] D. Dardari, P. Closas, and P. M. Djurić. Indoor tracking: Theory, methods, and technologies. *IEEE Transactions on Vehicular Technology*, 64(4):1263–1278, 2015.

[7] D. Dardari, E. Falletti, and M. Luise. *Satellite and terrestrial radio positioning techniques: a signal processing perspective*. Academic Press, 2011.

[8] F. Dovis. *GNSS interference threats and countermeasures*. Artech House, 2015.

[9] E. Kaplan and C. Hegarty. *Understanding GPS: principles and applications*. Artech house, 2005.

[10] G.-H. Lee, J. Jo, and C. H. Park. Jamming prediction for radar signals using machine learning methods. *Security and Communication Networks*, 2020, 2020.

[11] S. Liu and W. Deng. Very deep convolutional neural network based image classification using small training sample size. In *2015 3rd IAPR Asian conference on pattern recognition (ACPR)*, pages 730–734. IEEE, 2015.

[12] H. Mathis, P. Flammant, and A. Thiel. An analytic way to optimize the detector of a post-correlation fft acquisition algorithm. *Quadrature*, 1000:1, 2003.

[13] P. Misra and P. Enge. Global positioning system: signals, measurements and performance second edition. *Global Positioning System: Signals, Measurements And Performance Second Editions*, 206, 2006.

[14] R. Morales Ferre, A. de la Fuente, and E. S. Lohan. Jammer classification in gnss bands via machine learning algorithms. *Sensors*, 19(22):4841, 2019.

[15] T. J. O'Shea, T. Roy, and T. C. Clancy. Over-the-air deep learning based radio signal classification. *IEEE Journal of Selected Topics in Signal Processing*, 12(1):168–179, 2018.

[16] M. L. Psiaki and T. E. Humphreys. GNSS spoofing and detection. *Proceedings of the IEEE*, 104(6):1258–1270, 2016.

[17] E. Shafiee, M. Mosavi, and M. Moazedi. Detection of spoofing attack using machine learning based on multi- layer neural network in single-frequency gps receivers. *The Journal of Navigation*, 71(1):169–188, 2018.

[18] S. Sharma. Activation functions in neural networks. *Towards Data Science*, 6, 2017.

[19] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[20] M. Sun, Y. Qin, J. Bao, and X. Yu. Gps spoofing detection based on decision fusion with a k-out-of-n rule. *IJ Network Security*, 19(5):670–674, 2017.

[21] J. B.-Y. Tsui. *Fundamentals of global positioning system receivers: a software approach*, volume 173. John Wiley & Sons, 2005.

[22] A. D. Whalen. *Detection of signals in noise*. Academic press, 2013.

[23] Q. Yan, W. Huang, and C. Moloney. Neural networks based sea ice detection and concentration retrieval from gnss-r delay-doppler maps. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(8):3789–3798, 2017.