MIMO Beam Signature Detection for 5G based on Machine Learning

João Pedro Melquiades Gomes, Matheus Vilarim P. dos Santos, *Graduate Student Member, IEEE,* Edmar C. Gurjão, *Senior Member, IEEE*

***Abstract*—This paper explores applying deep learning techniques to enhance physical layer security in 5G networks by analyzing MIMO beam patterns. The study focuses on developing and evaluating three machine learning models—Denoising Autoencoder (DAE), Convolutional DAE, and Convolutional Neural Networks (CNNs)—to identify unique beam signatures caused by manufacturing imperfections in antenna arrays. Using the generated simulation datasets, the models are trained to distinguish devices based on their transmission beam patterns. The performance of the models is evaluated using metrics such as accuracy, precision, recall, and specificity. The results demonstrate the potential for using neural networks to implement robust security measures in 5G networks by leveraging intrinsic physical characteristics of antennas, providing a new layer of protection against unauthorized access.**

***Index Terms*—5G, Deep Learning, Cybersecurity, Physical layer.**

I. Introduction

The Technological-Scientific-Informational paradigm makes mass connectivity essential for contemporary human interactions, social relations, and economic activities. 5G networks represent the next stage of wireless broadband, offering higher data rates, lower latency, and increased bandwidth, with the potential to revolutionize smart cities, virtual reality, and the connectivity of IoT devices. However, the complexity and widespread use of software-defined systems in 5G networks introduce significant security vulnerabilities, threatening the privacy and reliability of transmitted data, thus necessitating more sophisticated security solutions [1].

Simultaneously, artificial intelligence (AI) has proven valuable in enhancing cybersecurity efforts, particularly in applications such as malware analysis, intrusion detection, and the security of energy systems. AI's ability to detect imperceptible patterns to humans and its capacity to automate the protection of large volumes of data make it a promising approach for addressing the security challenges posed by 5G networks [2].

Millimeter waves (mmWaves) play a crucial role in 5G technology, enabling high data transfer rates in cellular networks and facilitating the emergence of the Internet of Everything (IoE). As 5G networks evolve, they allow for the simultaneous connection of millions of devices, significantly expanding network scale. However, this advancement also raises security concerns, particularly in IoT networks, where spoofing attacks are commonly observed due to insecure practices, such as the continued use of default passwords on devices.

In light of these challenges, this work, divided into two parts, first aims to conduct a systematic review of the use of machine learning in cybersecurity applications for 5G networks, providing a comprehensive analysis of the current state of research in this field. Following this review, we contribute to the field by developing a project in a physical layer scenario. This paper uses Denoising Autoencoder (DAE) networks and Convolutional Neural Networks (CNNs) to authenticate IoT devices based on the physical characteristics of their antenna arrays. The goal is to evaluate the effectiveness of intruder detection in two scenarios: one involving static user equipment and the other involving devices capable of changing their position within the environment. Metrics such as accuracy, precision, recall, and specificity will be analyzed to assess the performance of the authentication system.

II. Related Works

To support the proposed solution, we reference relevant recent studies that apply AI to cybersecurity in 5G networks. Several works have explored deep learning for intrusion detection, jamming recognition, and beam management. Few of these focus specifically on physical-layer security in 5G systems. Recent research has demonstrated the dual nature of machine learning (ML) in 5G security - both as a potential vulnerability and a powerful defensive tool. Suomalainen et al. [3] provided a foundational analysis of this duality, examining how adversaries leverage sophisticated ML techniques while defenders simultaneously develop ML-based countermeasures. Building on this security paradigm, Sharma et al. [4] investigated the specific applications of ML and deep learning (DL) in securing 5G-enabled Industrial IoT systems, particularly focusing on threat detection in complex industrial environments. A comprehensive framework for understanding these emerging challenges was presented by Afaq et al. [5], who proposed a novel security architecture and identified critical research directions in ML-based 5G security. Further advancing this field, Almutairi [6] developed specialized deep learning solutions for traditional 5G networks and the emerging 5G-enabled Internet of Vehicles (IoV), providing concrete implementations of theoretical security frameworks. Sahni and Kaur systematically

J. P. M. Gomes (ORCID: https://orcid.org/0009-0006-4511-1320), M. V. P. dos Santos (ORCID: https://orcid.org/0000-0001-7708-9005), and E. C. Gurjão (ORCID: https://orcid.org/0000-0001-9694-3668) are with the Federal University of Campina Grande (UFCG), Campina Grande, Paraiba, Brazil (e-mail:joao.melquiades,matheus.vilarim,ecg@ee.ufcg.edu.br)

This work was carried out with the support of CNPq, National Council for Scientific and Technological Development - Brazil, within the program CNPq/MCTI/SEMPI No. 56/2022.

Submission: 2024-10-24, First decision: 2025-01-23, Acceptance: 2025-10-24, Publication: 2025-10-29.

Digital Object Identifier: 10.14209/jcis.2025.9

mapped the evolving threat landscape [7] and conducted a comprehensive review correlating specific threat vectors with corresponding ML-based mitigation strategies. A significant breakthrough in proactive security came from Mozo et al. [8], who introduced B5GEMINI, an AI-powered digital twin network model that enables predictive vulnerability detection through sophisticated simulation techniques. This progression of research underscores the critical role of machine learning in modern 5G security frameworks, directly informing the potential contribution of our approach to MIMO beam signature detection, thus enhancing physical layer security in 5G.

III. Concepts

According Equation 1, the received power P_{Rx} is function fo the transmitted power P_{Tx}, gain of the receiver and transmitted antennas G_{Rx} and G_{Tx}m the wavelength λ, distance between the transmitter and receiver r [9].

$$P_{rx} = \frac{P_{Tx}}{4\pi r^2}\frac{\lambda^2}{4\pi}G_{Rx}G_{Tx}. \quad (1)$$

In a 5G network, devices may operate at high-frequency bands (above 24 GHz), and the signal power decreases significantly compared to low-frequency transmissions. One possible solution to compensate for this signal strength reduction is to increase the gain of the transmitter and/or receiver by:

- Increasing the size of the antennas: While enhancing gain, it also makes the radiated signal more directional, and if radiation is required in multiple directions, mechanical adjustments are necessary to reposition the entire antenna structure, rendering this option impractical;
- Increase the number of antennas to form an array: the gain G_{Rx} or G_{Tx} becomes the array gain. Additionally, the radiated signal will be focused on a beam, and the beam direction can be altered through digital signal processing by adjusting the phase shift between each element in the array.

Antenna arrays have become a viable solution for high-frequency operations, as they mitigate signal power loss caused by the frequency increase while simultaneously reducing the required array size and eliminating the need for mechanical adjustments to radiate in multiple directions. Consequently, in an array with identical elements, five control parameters can be adjusted to shape the array's beam pattern [10]:

- Geometric configuration of the array (This work used a rectangular array);
- Relative distance between elements;
- Phase shift between elements. This phase shift allows the array's beam to be steered in a specific direction;
- The individual pattern of each element.

Therefore, the beam pattern can be defined as a function of two factors: the individual pattern of an element, $E.P(\theta, \phi)$, and the contributions from the array, known as the Array Factor ($A.F(\theta, \phi)$), given by:

$$A.P(\theta, \phi) = E.P(\theta, \phi) \times A.F(\theta, \phi) \quad (2)$$

where θ and ϕ are the azimuth and elevation angles, respectively. In this work, only changes to the array factor were made by adjusting the spacing between each element to obtain a unique signature for each array. It is shown the individual pattern in Fig. 1, the array factor in Fig. 2, and the combined beam pattern in Fig. 3. These images were generated for a $6x6$ array designed to operate at 28 GHz.

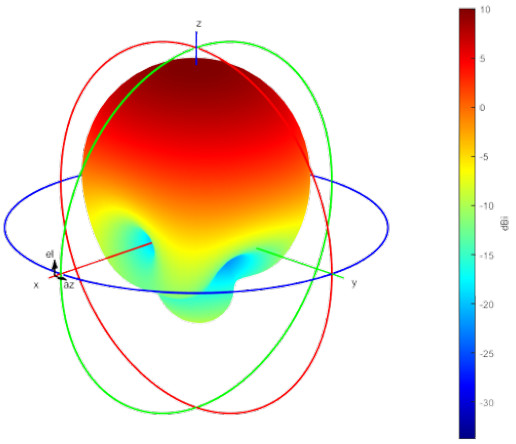

Fig. 1. Individual pattern of each element.

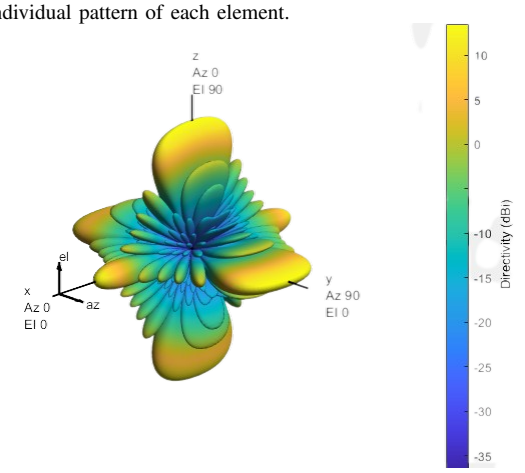

Fig. 2. Array factor.

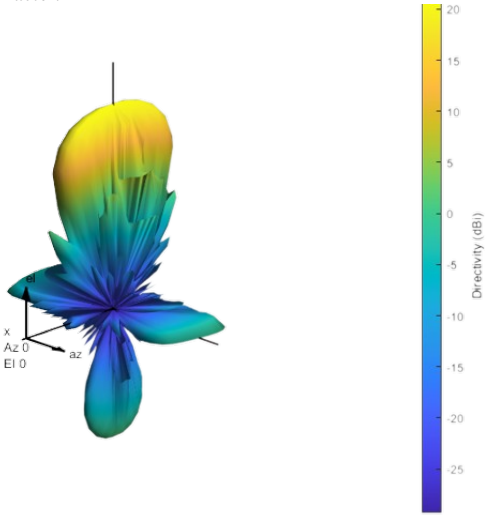

Fig. 3. Beam pattern.

A. Deep Learning

Deep learning is a particular area within the field of machine learning, composed of neural network models. In Fig. 4, the main components of this structure are shown. A linear combination of inputs x_i, weighted by w_i, is compared to a threshold, and the result is passed through an activation function.

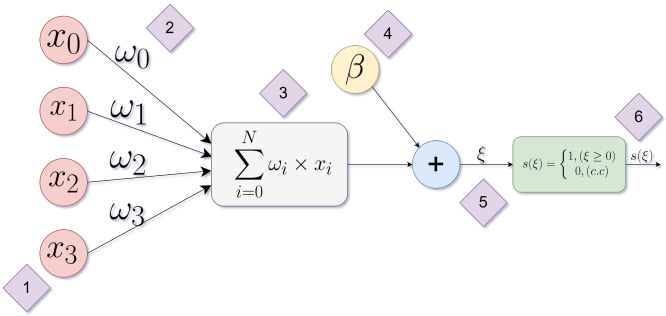

Fig. 4. Perceptron Architecture.

The weights are obtained in the training stage using a dataset of cases.

Convolutional Neural Networks (CNNs) are a class of artificial neural networks that utilize convolutional operations. Convolution is a mathematical process that enables extracting important features from images or signals, such as edges, textures, and shapes. CNNs are extensively employed in computer vision tasks, including object recognition, image classification, and pattern detection [11].

CNNs represent an advancement over fully connected layers, the traditional neural network layers where each neuron is connected to all the neurons in the preceding layer. In contrast, CNNs employ convolutional layers that apply filters to specific regions of an image, enabling the network to learn local, translation-invariant features. This approach significantly reduces the number of parameters in the network, enhancing its efficiency and ability to process high-resolution images. To fully appreciate the advantages of CNNs, it is essential to define two key principles on which these models are based: translation invariance and locality [12].

An autoencoder is a neural network designed to learn how to replicate its input in the output by using a hidden layer to represent the content of the input [11]. Typically, its structure is divided into two components: the encoder, which maps the input to the hidden layer ($h = f(x)$), often compressing the input during the encoding process, and the decoder, which reconstructs the input from the hidden layer ($r = g(h)$), decompressing it in the process. As a result, the reconstruction does not aim to be an exact copy of the input since there are inherent losses throughout the encoding process. Instead, the model focuses on selecting the input's most important features, learning to reconstruct data similar to those used during training.

Autoencoders are traditionally used for dimensionality reduction and feature extraction [13]. However, more recent theoretical connections between autoencoders and latent variable models have made them essential for generative modeling.

1) Performance Metrics: To compare the different models, four metrics were selected from the various existing deep learning measures: accuracy, precision, recall, and specificity. These metrics are defined in a binary classification model, where the labels are **Positive** or **Negative**.

- **Accuracy** - This metric can be defined as:

$$Acc = \frac{TP + TN}{TP + FP + TN + FN}$$

  where:
  - $TP \rightarrow$ True Positives, meaning correctly predicted true values;
  - $FP \rightarrow$ False Positives, meaning false values predicted as true;
  - $TN \rightarrow$ True Negatives, meaning correctly predicted false values;
  - $FN \rightarrow$ False Negatives, meaning true values predicted as false;

  The main purpose of accuracy is to provide an overall view of the model's success. However, it does not indicate whether the model predicts more true or false values correctly.
- **Precision** - This can be defined as:

$$Precision = \frac{TP}{TP + FP}.$$

  This metric indicates the percentage of true values predicted that are actually true. For a perfect model, the Number of false positives will be zero, and the precision value will be 1.
- **Recall** - This can be defined as:

$$Recall = \frac{TP}{TP + FN}.$$

  This metric indicates the percentage of true values predicted relative to all true values.
- **Specificity** - This can be defined as:

$$Specificity = \frac{TN}{TN + FP}.$$

  This metric indicates the opposite of precision. It shows the percentage of values predicted as false that are actually false.

These indicators can be aggregated into a confusion matrix with dimensions 2×2 for binary classification, and it is designed to facilitate quick analysis of a model's performance. The main diagonal of this matrix consists of true positives and negatives, with high values indicating that the model has correctly classified most of the data. A color gradient is often applied to this representation, with higher values associated with more intense colors within the gradient.

IV. Problem Statement

Among the important characteristics of 5G networks, and of particular interest to this work, is beam management, which is employed to establish and maintain transmission and reception beams [14]. These beams, in turn, are generated by MIMO antennas, which, although typically manufactured in series, may exhibit small differences, such as millimeter-scale

variations in size, or have their radiation patterns altered due to the presence of obstructions, such as bulkheads, within the equipment.

Due to the modifications, two MIMO systems with the same configuration and receiving identical parameter settings may generate different radiation patterns. While these variations might not significantly affect performance, they can create a distinct 'signature' in the radiation patterns.

Determining a user's transmission signature can have applications ranging from user authentication to enhancing security. This is particularly useful when it is necessary to unambiguously confirm that a transmission was perfomed by a specific device, such as in the case of signal jammers.

In 5G, beam management consists of an initial scanning phase, when multiple beams are transmitted in different directions, followed by a beam measurement phase, where the UE or *gNodeB* performs power measurements to evaluate the intensity of the radiated beams and refine the receiving beam. In this process, the transmitter radiates a single beam while the receiver scans the space using multiple beams to establish the optimal direction. This process is also based on the measurement of CSI-RS or SRS [15].

In this sense, there are several moments in which the signature of a transmission beam can be evaluated, and in this work, neural networks will be used to determine the signature of a transmission.

V. Methodology

Considering an architectural view of 5G, the simulated scenario consisted of: UE sends a Random Access Channel (RACH) preamble request to the gNodeB, the station will receive this signal radiated by the UE elements, and based on the beam pattern transmitted by the UE, a model executed in the RAN (Radio Access Network) will identify whether the signal is coming from a UE belonging to the network or is an invading device. Fig. 5 presents a view of the models running on the RAN, one in near real-time to identify connected devices and another in non-real-time to learn the radiation pattern of new legitimate users authenticated on the network. The models could be seen as applications running in a RAN intelligent controller environment.

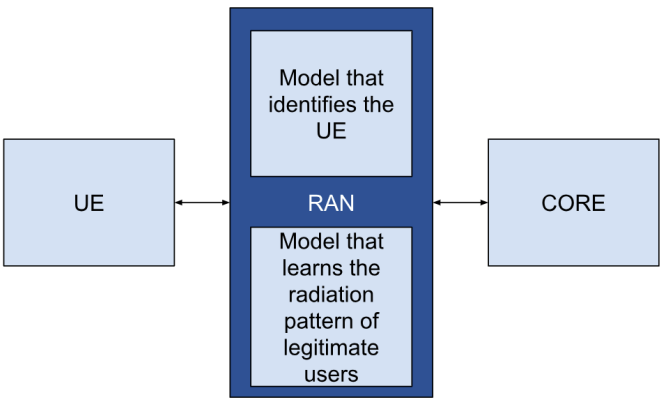

Fig. 5. Solution View Applied to 5G Architecture.

A dataset of antenna pattern transmissions was created via simulation to evaluate the use of neural networks to detect intrusions through antenna patterns. A portion of the generated data was then assessed to assess the detection performance.

We propose a hybrid model architecture deployed at the RAN level to detect intrusions through antenna pattern analysis. This architecture consists of two components operating on different time scales. The first is a near real-time model responsible for recognizing beam signatures of UEs already known and authenticated in the network. The second is a non-real-time learning model that registers new legitimate devices by learning their radiation patterns offline before allowing them into the trusted pool. This separation enables adaptive security with minimal latency impact.

While traditional authentication occurs at higher protocol layers, our system adds a layer of physical layer validation based on inherent hardware characteristics, which is particularly valuable in defending against sophisticated spoofing attacks where intruders may possess valid credentials but cannot replicate the subtle imperfections in beam patterns unique to each device. By analyzing these patterns at the physical layer, our approach acts as a biometric validation step that is protocol-independent and robust against identity theft or cloning.

This solution is embedded into a simulated 5G communication flow: during the RACH (Random Access Channel) phase, the gNodeB receives the transmitted signal and extracts the beam pattern, which the detection model evaluates. This scenario, illustrated in Fig. 5, contextualizes how the beam pattern is used as a biometric-like feature for device validation.

The system relies on the idea that manufacturing variations in antenna arrays create subtle but consistent differences in beam patterns unique to each device. Autoencoders are trained exclusively on the patterns of known devices. When an unknown (potentially malicious) device attempts communication, its beam pattern deviates significantly from the learned distribution, resulting in high reconstruction error. This anomaly is flagged as a potential intrusion.

A. Data Set Construction

The datasets were generated using MATLAB's 5G Toolbox for simulating an environment for modeling physical layer impairments such as white noise, amplifier nonlinearity (distortion), I/Q imbalance, phase noise, and RF filtering. These impairments were introduced to the beam patterns to emulate real-world signal distortions and provide a robust foundation for training and evaluating the models.

MATLAB scripts create various antenna array configurations by varying specific parameters to generate different beam patterns. These patterns were analyzed, and a preliminary assessment was conducted to determine which antenna parameters would be employed in the subsequent development stages. This assessment considered the influence on the beam pattern, the trade-off between the computational demands for simulating these arrays, and the project completion time.

Subsequently, a micro 5G environment was implemented within MATLAB using the 5G Toolbox, encompassing waveform generation, a transmitting antenna array, a channel, a

receiving antenna array, and a power measurement mechanism. This system generated three databases, each containing beam patterns of four devices equipped with a 6x6 antenna array with varying transmission angles in both azimuth and elevation.

1) *StaticPosAllUEs_Dataset*: measurements taken for a single UE position, modifying only the UE for each set of measurements. This database aims to verify whether the detection is based solely on beam patterns and not on variations caused by the distance or transmission angle of the antennas;
2) *StaticPosSingleUE_Dataset*: measurements taken for the four devices at different locations, albeit without position variation during the process;
3) *DynamicPos_Dataset*, includes measurements taken with the four devices varying their positions during the measurement process, thus enabling the construction of a somewhat more robust model.

Each of these databases was divided into two parts: one for training (80% of the data) and one for testing (20% of the data) the models.

The selected models DAE, ConvDAE, and CNN, were chosen for their complementary characteristics in handling structured input data. DAEs are used in anomaly detection tasks because they can learn compact representations and highlight deviations from known input distributions. ConvDAEs enhance this capability by leveraging spatial locality, making them well-suited for reconstructing 2D beam pattern structures. CNNs on the other hand, are optimized for classification and allow the system to learn decision boundaries based on the extracted features explicitly. These models have all demonstrated effectiveness in physical-layer security applications in prior studies. However, few works apply them in tandem to beam pattern analysis for MIMO-based device authentication. Our horizontal evaluation of these models addresses a gap in comparing reconstruction-based and classification-based deep learning for this purpose.

Three artificial intelligence models were developed in Python, specifically the TensorFlow development framework. The trained models were:

- DAE - Denoising Encoder;
- Convolutional DAE;
- CNN (Convolutional Neural Network).

These three models were then employed to make predictions on the test portions of the databases, thereby comparing the metrics of each model. The choice between DAE, ConvDAE, and CNN affects performance, feature extraction, and complexity. As a fully connected model, DAE captures global dependencies but lacks spatial awareness, limiting its ability to preserve localized patterns. ConvDAE uses convolutional layers to exploit spatial correlations, enhancing generalization and making it more effective for distinguishing device signatures. CNNs, optimized for classification, extract hierarchical features and define decision boundaries but require extensive labeled data. While CNNs excel in classification, ConvDAEs balance feature extraction and anomaly detection, whereas DAEs may struggle with high-dimensional spatial data.

The CNN in the article consists of three convolutional layers with 32, 64, and 64 filters, followed by two dense layers with 8 and 2 neurons, resulting in approximately 155,648 operations per forward pass due to its convolutional operations. The ConvDAE uses an encoder with 16, 16, and 8 filters and a decoder with 8, 16, and 16 filters, followed by a final convolutional layer, totaling around 38,016 operations per forward pass, primarily from its stride and transposed convolutions. The DAE features dense layers with 256, 128, 64, 32, and 16 neurons in the encoder and 32, 64, 128, and 256 neurons in the decoder, leading to approximately 98,304 operations per forward pass, driven by its fully connected layers. These calculations highlight the trade-offs between convolutional and dense architectures in terms of computational complexity.

The CNN, DAE, and ConvDAE models were trained using the Adam optimizer with a learning rate of 0.001. The CNN was trained with a batch size of 32 for 10 epochs, while the DAE and ConvDAE were trained for 50 epochs. The loss functions used were sparse categorical crossentropy for the CNN and mean squared error for both autoencoders. Tab. I summarizes these hyperparameters to improve clarity and reproducibility.

The evaluation metrics employed—accuracy, precision, recall, and specificity—were chosen under the assumption of a balanced dataset. Our simulation framework includes equal representation of legitimate and intruder devices (50% each), allowing these metrics to provide fair insights into model performance.

To translate the reconstruction error into a classification decision, we compute the Mean Squared Error (MSE) between each sample's input and the reconstructed output. During training, a statistical threshold is calculated using the mean reconstruction error across all training samples from legitimate devices, augmented by one standard deviation. A test sample is classified as intruders if its reconstruction error exceeds this threshold. This approach assumes a compact error distribution for known devices, such that anomalous signals result in significantly larger errors.

VI. Results

This section details the results obtained with the databases created in *MATLAB*. All graphs were generated in *Python* with the *Matplotlib* and *Seaborn* libraries, and the models were trained and tested using the *Tensorflow* development infrastructure. All files used in this work were made available in a public repository[1] to increase reproducibility and facilitate future work.

While the computational complexity of each model was initially described in the Methodology, its impact on practical deployment has not yet been discussed. We now highlight that although the CNN achieved the highest accuracy in all scenarios, it also demanded the highest number of operations per forward pass (155,648). In contrast, the ConvDAE required only 38,016 operations while achieving competitive detection rates. This trade-off between performance and computational

[1]https://github.com/LAPSI-DEE-UFCG/5G-PHY-Security

TABLE I
SUMMARIZED HYPERPARAMETERS USED.

Model	Optimizer	Learning Rate	Batch Size	Epochs	Loss Function	No. of Neurons	Computational Complexity (Ops)
CNN	Adam	0.001	32	10	Sparse Categorical Crossentropy	170	155,648
DAE	Adam	0.001	Full Batch	50	Mean Squared Error	960	98,304
ConvDAE	Adam	0.001	Full Batch	50	Mean Squared Error	81	38,016

efficiency is critical in environments with constrained processing power and energy, such as edge nodes in 5G infrastructure. Moreover, while the DAE provided strong reconstruction capability, its use of dense layers results in a significantly higher parameter count and resource demand, making it less practical for lightweight implementations. Thus, ConvDAE presents an optimal balance for edge-level deployment.

1) Device Beam Patterns: As mentioned in the methodology, three distinct datasets were generated, each element containing a measurement vector consisting of 256 samples. These samples represent variations in the transmission angle in both azimuth and elevation. The variations were made within a range of $\pm 15^o$ from the direct incidence angle between the transmitting and receiving devices.

Fig. 6 presents the beam patterns for the *StaticPosAllUEs_Dataset*. Each graph in the figure contains 1,000 overlapping patterns. A notable similarity can be observed among the patterns for each device, with slight differences caused solely by manufacturing imperfections in the antennas. These variations arise even though all devices were positioned in the same location for power measurements. In Fig. 7, the antenna patterns for each device are displayed in a three-dimensional view. In this representation, the coordinates correspond to the transmission angles, with reference to the direct incidence angle (point $(0, 0, P)$, where P is the measured power).

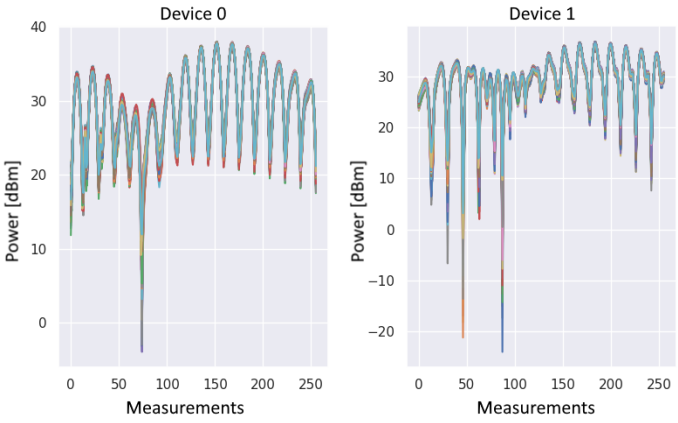

Fig. 6. Beam patterns - *StaticPosAllUEs_Dataset.*

The second database, obtained through power measurements with static UEs but in different positions, has a set of beam patterns that have more differences between devices, but the measurements are still very similar for the same device. It is possible to observe this similarity by analyzing the overlap of the curves of the same UE in Fig. 8. This overlap gives the impression that it is just one curve with a thicker contour.

Finally, the measurements that resulted in the third database, the *DynamicPos_Dataset*, are much more distinct, both between devices and measurements of the same device. This is because, in this scenario, the UEs had their positions varied throughout the measurements, thus recording the beam patterns for different positions. This makes training more difficult, but it is also responsible for the greater robustness of the recognition models. In Fig. 9, all beam patterns from this database can be seen, and in Fig. 10, a pattern for each device is shown in a three-dimensional view.

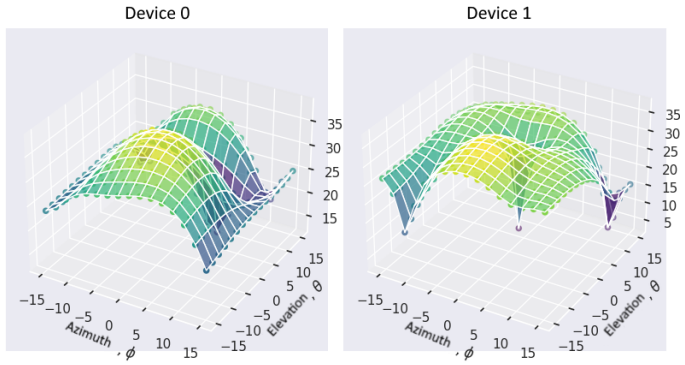

Fig. 7. Three-dimensional beam patterns - *StaticPosAllUEs_Dataset.*

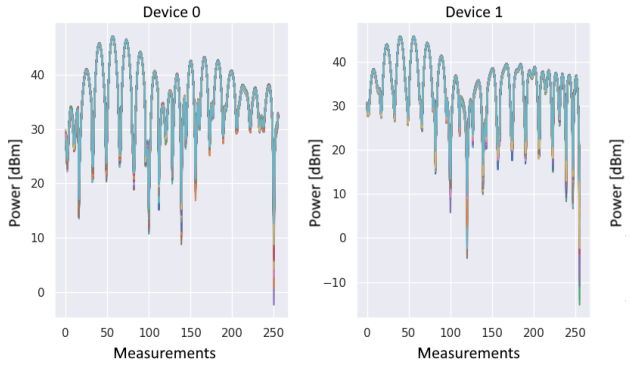

Fig. 8. Beam patterns - *StaticPosSingleUE_Dataset.*

2) Model Descriptions: The first trained model was the Convolutional Neural Network (CNN). These networks perform well for image classification, so the database vectors consisting of the function of azimuth and elevation were represented as images. It was constructed with three convolutional layers and one dense layer, having an input of $(16x16)$ and a binary output, classifying the device as *True* or *False*. This model has a practical disadvantage; for its training, it was necessary to use antenna patterns that did not belong to the network. In a real application, these measurements would have to be created and not measured, as a real system would only have measurements from authenticated devices.

The second model was a Deep Autoencoder (DAE). The main objective of this model is to reconstruct the input at

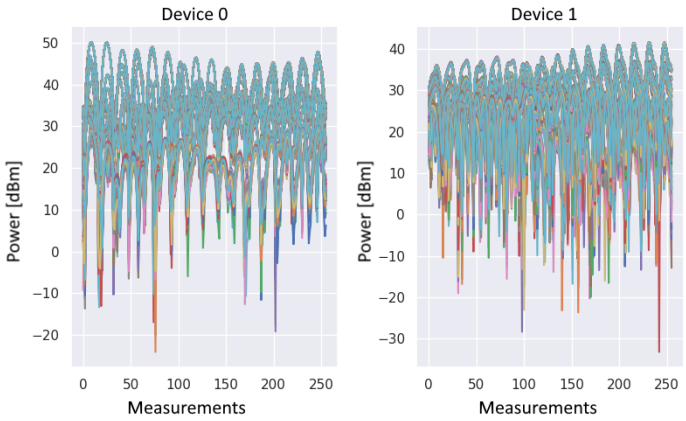

Fig. 9. Beam patterns - *DynamicPos_Dataset*.

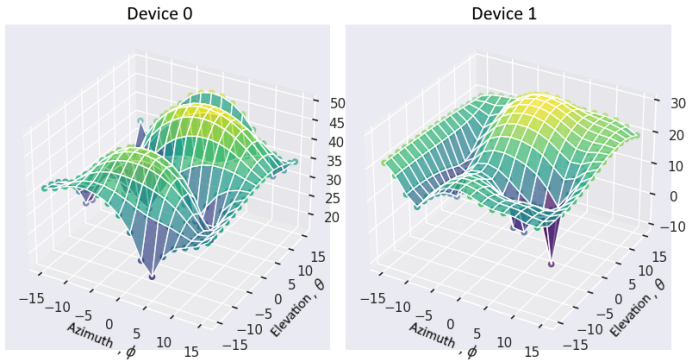

Fig. 10. Three-dimensional beam patterns - *DynamicPos_Dataset*.

the output. To do this, it compresses the input data and subsequently decompresses it at the output, trying to reconstruct the information lost during compression. In the anomaly detection scenario, a model trained to reconstruct the beam pattern of a device belonging to the network will not be able to reconstruct patterns from invasive devices. Thus, based on the reconstruction error, it is possible to identify whether the UE is an invader or belongs to the network.

Finally, the last implemented model was a convolutional DAE. Its objective is the same as the conventional DAE. However, its implementation involves creating convolutional layers instead of fully connected ones. Thus, it is helpful for training image reconstruction models. By converting the vectors back to a size of $16x16$, it is possible to use this approach and try to extract more characteristics from the patterns. Fig. 11 presents the architecture of this network.

With the neural networks defined, the results and comparisons of the metrics for each of the databases will be presented next.

3) Comparison of Metrics - StaticPosAllUEs_Dataset: Only five epochs were necessary for CNN network training. This is because the patterns have distinct differences, allowing the model to identify patterns in the generated images quickly. The DAE models, however, required 25 epochs for effective training. Furthermore, the convolutional neural network generally demonstrated superior results in three of the four metrics analyzed: precision, accuracy, and recall.

The comparison between autoencoders can be made in three aspects: Reconstruction of a beam pattern from an authenticated device, reconstruction of a beam pattern from an intruder device, and the error threshold value used to identify whether a UE belongs to the network. This threshold is calculated based on the mean error value of the training samples, added to the standard deviation of these errors. Any value above this threshold is sufficient to consider a device an intruder.

Regarding the reconstruction of beam patterns, in Fig. 12, it is possible to observe that both DAEs were able to sufficiently reconstruct the input data, with the conventional DAE presenting a better reconstruction compared to the convolutional DAE. In Fig. 13 and 14, one can see the attempt to reconstruct a beam pattern obtained from an intruder for both models, along with the reconstruction error highlighted in red.

Up to this point, comparing the DAEs with the convolutional neural network is impossible, as the latter does not aim to reconstruct the data at the output. However, the metrics mentioned in the previous section were compared based on the results with the portion of the database dedicated to testing. Fig. 15 shows these values. The CNN presented unitary values for all measures, as each device's antenna patterns are practically identical. It is essential to observe that the specificity was unitary for all models, meaning all intruder devices were detected, and there were no false positives.

Finally, the confusion matrix for the three models can be seen in Fig. 16. This matrix combines the four values used to calculate the metrics, such that a well-trained model will have high values on the diagonal of the confusion matrix, with the remaining values being as low as possible. In the case of DAEs, 77 and 78 false negatives were predicted for the ConvDAE and DAE, respectively. This reflects a recall of about 80%, which is acceptable for this model, given that the most important security aspect is blocking intruders.

4) Comparison of Metrics - StaticPosSingleUs_Dataset: For this database, CNN also presented better precision, accuracy, and recall results. The comparison of the reconstruction of the two DAEs can be seen in Fig. 17, 18 and 19. In this case, the conventional DAE presented slightly better results than the convolutional one, which was similar to the previous database.

Regarding the four metrics, the convolutional neural network still presented better results than the other models. It is important to remember that the number of epochs has not been altered, with all models being trained under the same conditions. In Fig. 20, it is possible to see that the results were very similar to the previous database.

Finally, the confusion matrices for the three models can be seen in Fig. 21. Here, the DAEs presented slightly better values, with 59 and 61 false negatives predicted for the ConvDAE and DAE, respectively, reflecting a recall of about 84%.

5) Comparison of Metrics - DynamicPos_Dataset: For this database, the training was more challenging. The CNN did not achieve perfect results within 5 epochs, and the results of the Deep Autoencoders were unsatisfactory. In Fig. 22, 23 and 24, it is possible to observe how the DAE models

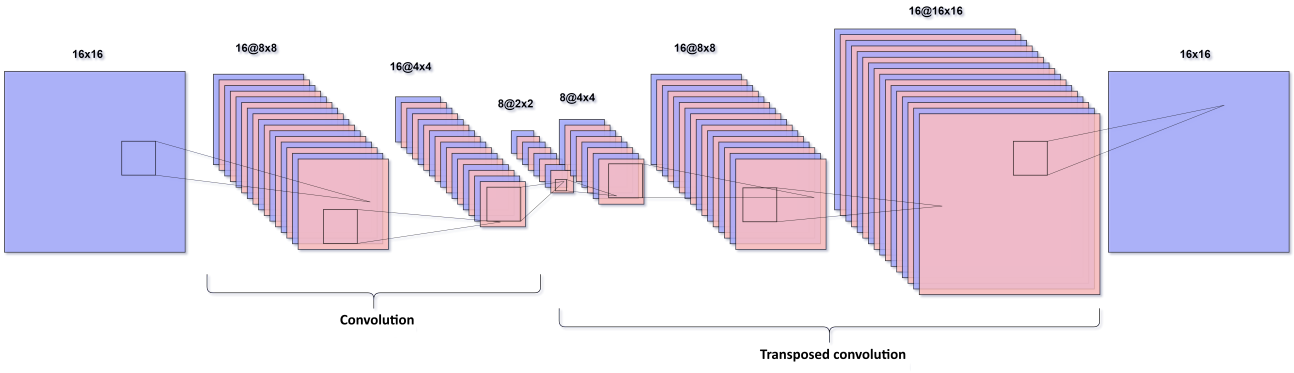

Fig. 11. Convolutional DAE Architecture.

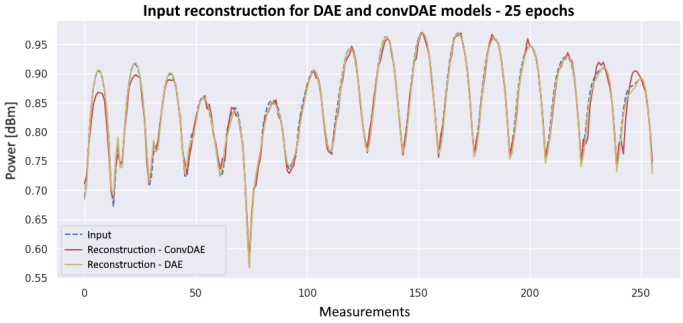

Fig. 12. Reconstruction of an authenticated beam pattern - *StaticPosAllUEs_Dataset*.

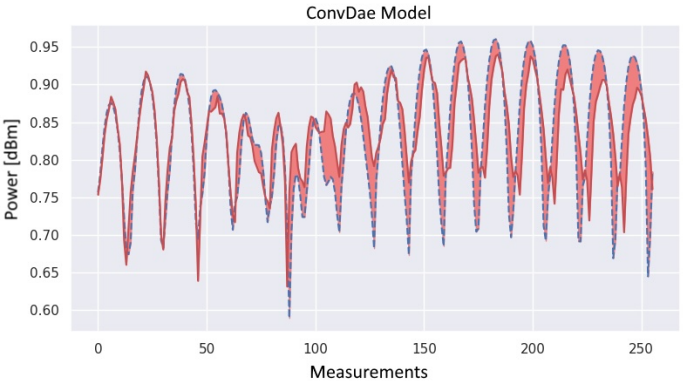

Fig. 13. ConvDae - Reconstruction of an intruder beam pattern - *StaticPosAllUEs_Dataset* with 25 epochs.

attempted to reconstruct the beam pattern of the devices. Additionally, Fig. 25 shows the metrics with lower values than the tests conducted with the previous databases. Finally, the confusion matrix in Fig. 26 demonstrates how the number of false positives was high, rendering the detection of intruders ineffective.

The CNN was retrained with 10 epochs to improve the results, and the DAE models were trained with 60 epochs. This led to a considerable improvement in recognizing all models, presenting results similar to the first two databases. The reconstruction of authenticated device patterns for this new training can be seen in Fig. 27. The metrics can be

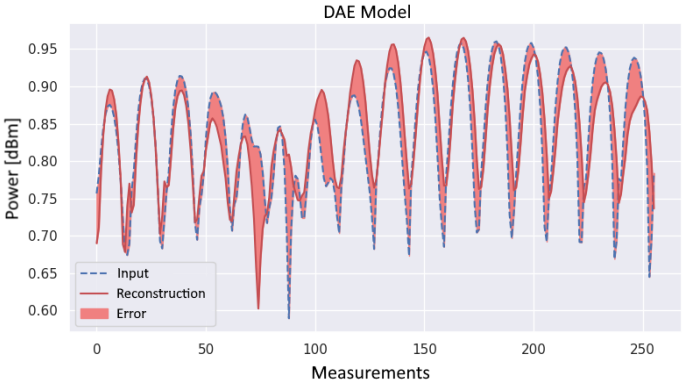

Fig. 14. DAE - Reconstruction of an intruder beam pattern - *StaticPosAllUEs_Dataset* with 25 epochs.

observed in Fig. 30, and the confusion matrix in Fig. 31.

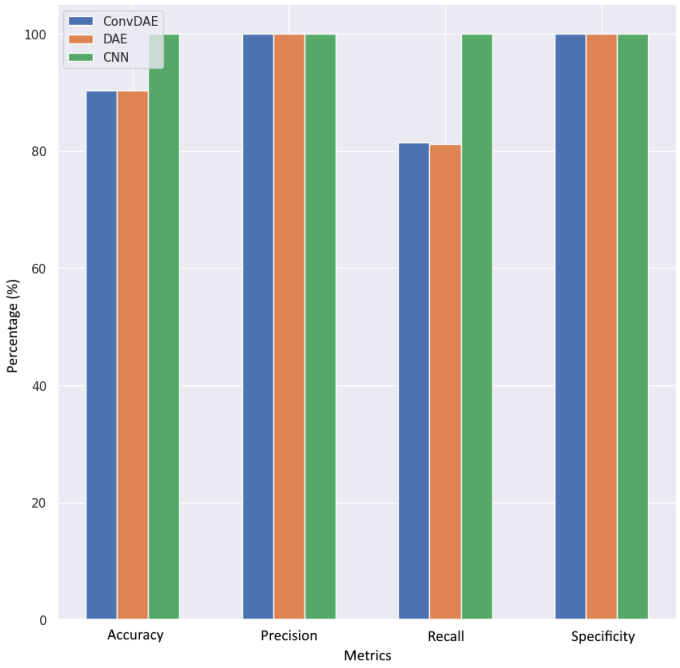

Fig. 15. Metrics for the three models - *StaticPosAllUEs_Dataset*.

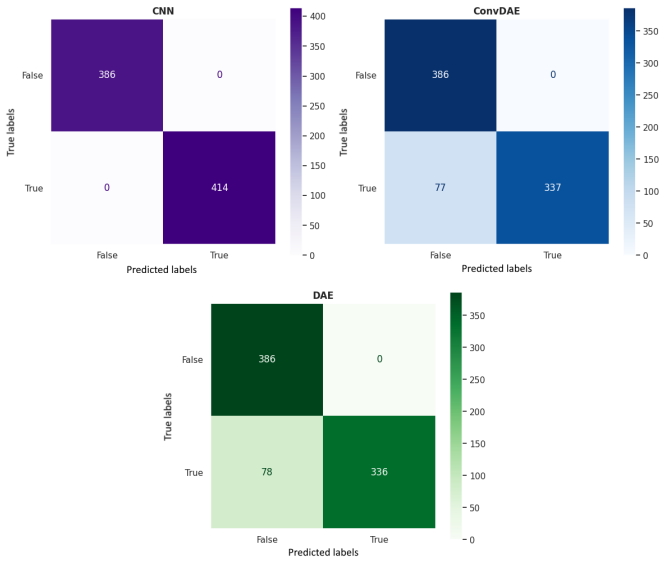

Fig. 16. Confusion matrix for the three models - *StaticPosAllUEs_Dataset*.

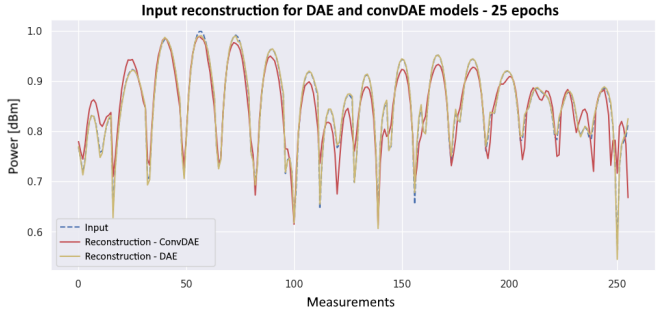

Fig. 17. Reconstruction of an authenticated beam pattern - *StaticPosSingleUE_Dataset*.

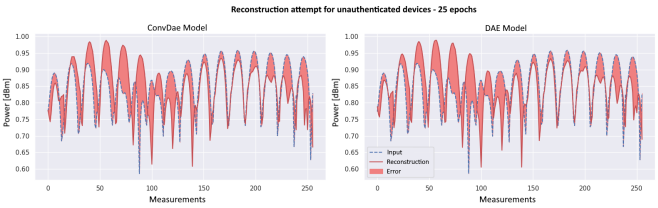

Fig. 18. Reconstruction of an intruder beam pattern - *StaticPosSingleUE_Dataset* for ConDae.

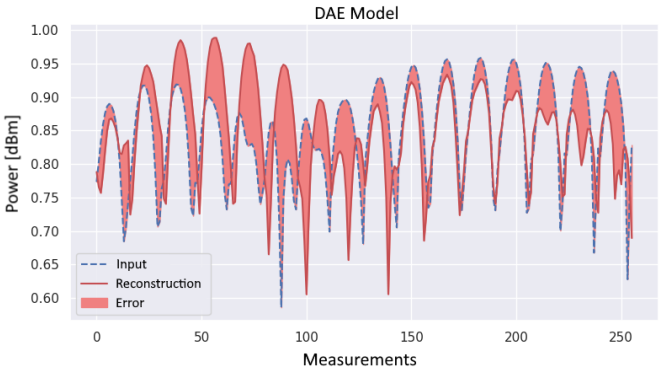

Fig. 19. Reconstruction of an intruder beam pattern - *StaticPosSingleUE_Dataset* for DAE.

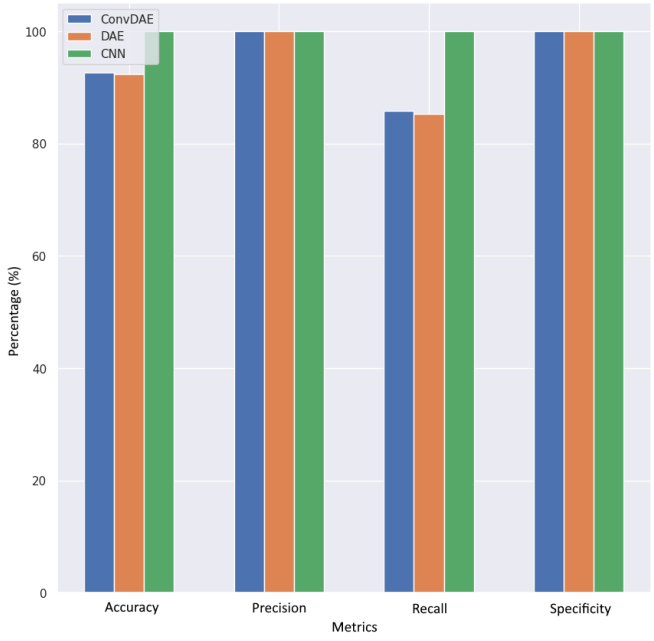

Fig. 20. Metrics for the three models - *StaticPosSingleUE_Dataset*.

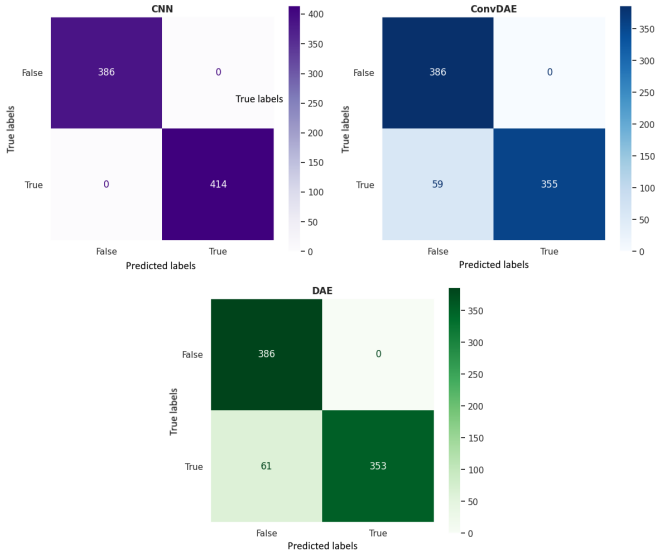

Fig. 21. Confusion matrix for the three models - *StaticPosSingleUE_Dataset*.

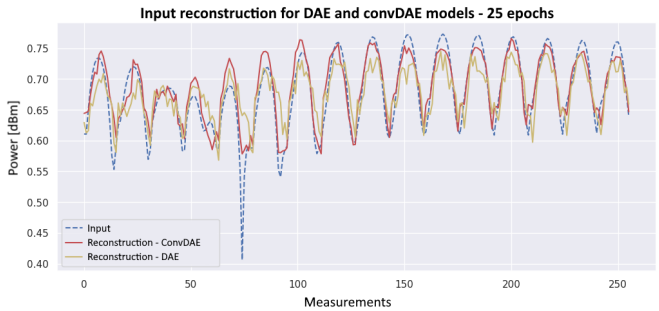

Fig. 22. Reconstruction of an authenticated beam pattern - *DynamicPos_Dataset*.

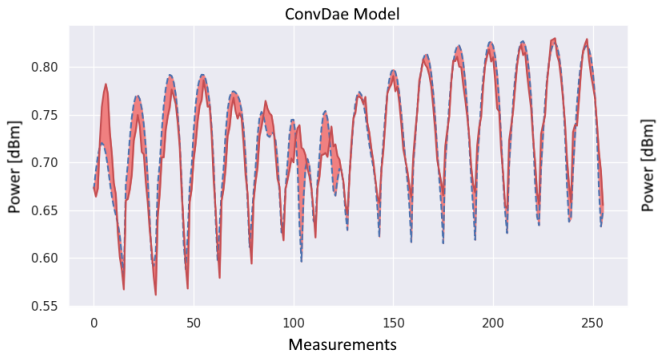

Fig. 23. Reconstruction of an intruder beam pattern - *DynamicPos_Dataset*.

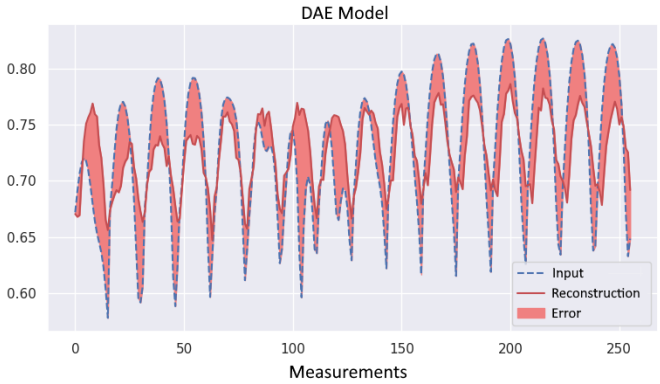

Fig. 24. Reconstruction of an intruder beam pattern - *DynamicPos_Dataset*.

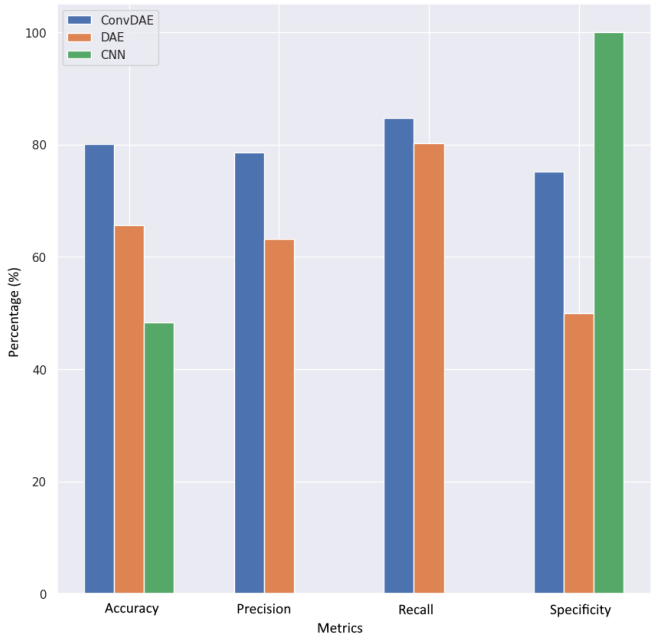

Fig. 25. Metrics for the three models - *DynamicPos_Dataset*.

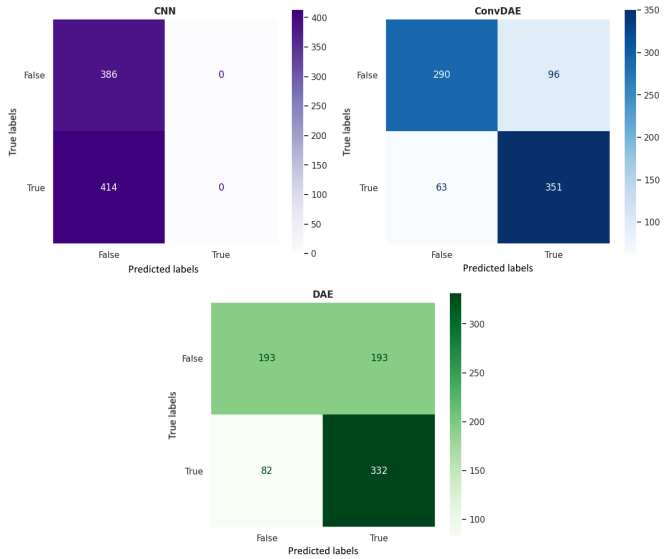

Fig. 26. Confusion matrix for the three models - *DynamicPosSingleUE_Dataset*.

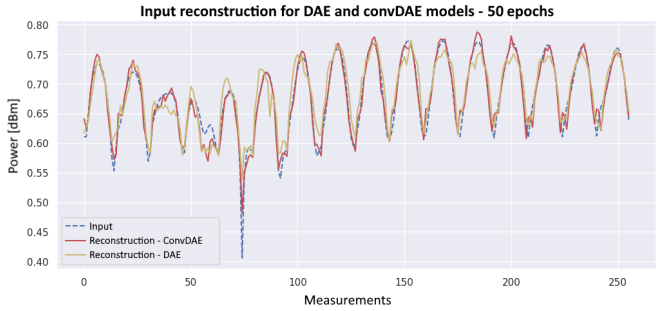

Fig. 27. Reconstruction of an authenticated beam pattern - *DynamicPos_Dataset* - Retraining.

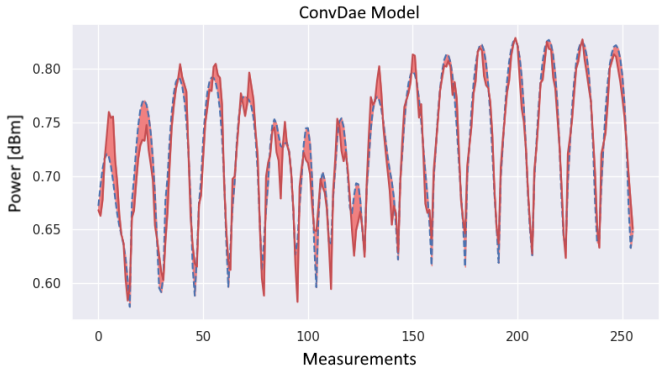

Fig. 28. Reconstruction of an intruder beam pattern - *DynamicPos_Dataset* - Retraining for ConvDae.

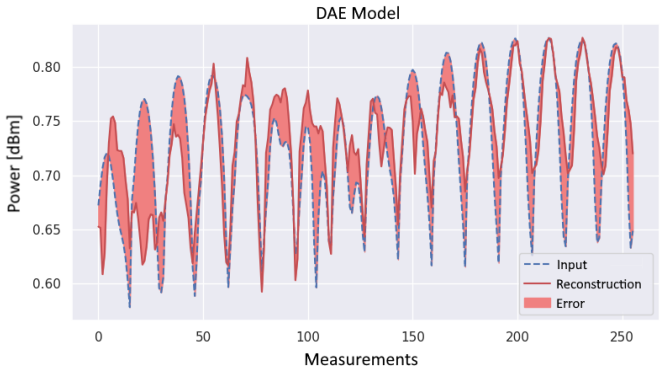

Fig. 29. Reconstruction of an intruder beam pattern - *DynamicPos_Dataset* - Retraining for DAE.

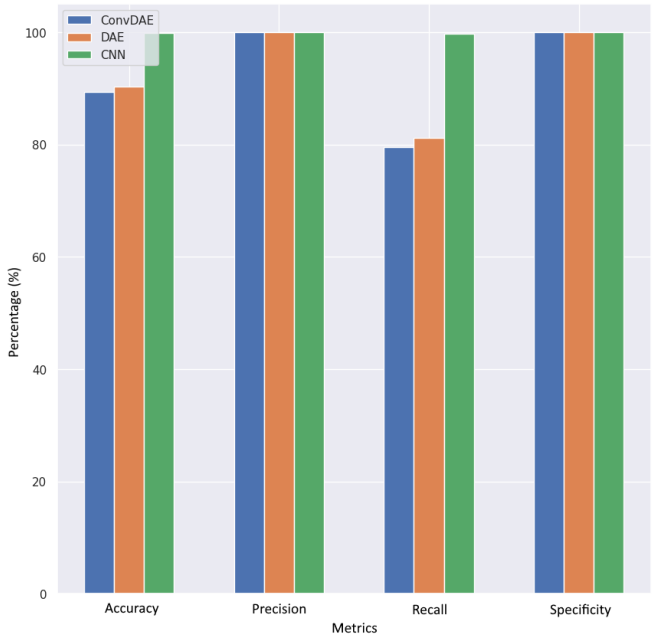

Fig. 30. Metrics for the three models - *DynamicPos_Dataset* - Retraining.

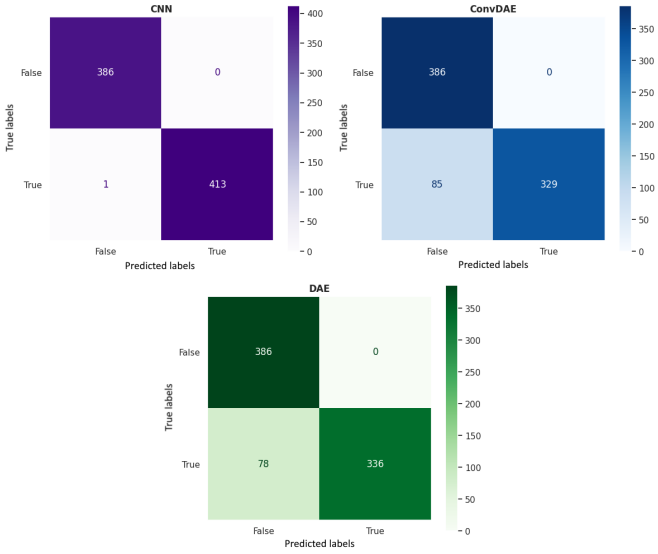

Fig. 31. Confusion matrix for the three models - *DynamicPosSingleUE_Dataset* - Retraining.

To verify the presence of overfitting, graphs relating to the training and validation errors were plotted, as can be seen in Fig. 32 and 33. The fact that the validation error curves do not increase and remain close to the training error curves indicates that the model is not overfitting the training data (i.e., there is no evident overfitting). In overfitting situations, we expect to see the validation error increase while the training error decreases.

The results presented in Tab. II highlight the complementary roles of the proposed models within the architecture. While the reconstruction-based models (ConvDAE and DAE) achieved a recall of approximately 80%, the classification model (CNN) reached a perfect recall of 100%. This demonstrates that although reconstruction is effective for anomaly detection, classification provides higher reliability for identifying malicious behaviors. Therefore, both approaches must coexist in the near-real-time and non-real-time layers of the open RAN 5G architecture, ensuring a more robust and comprehensive detection framework.

6) Areas for Improvement: Based on the results, the following points for improvement in the identification process can be listed:

- Generation of larger and more diverse databases for a scenario closer to reality, in which devices can occupy various positions;
- Obtaining received power levels from physical measure-

TABLE II
COMPARISON OF THE PERFORMANCE OF TRAINED MODELS

Metrics	ConvDAE	DAE	CNN
Accuracy	0.87	0.89	1.00
Precision	0.95	1.00	1.00
Recall	0.80	0.79	1.00
Specificity	0.95	1.00	1.00

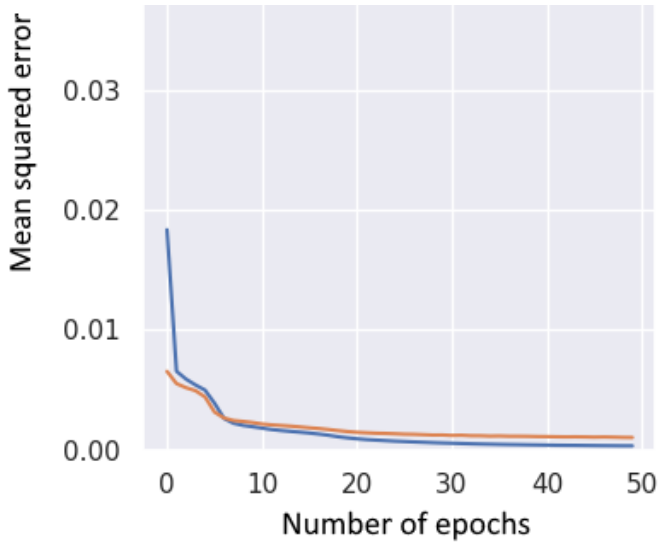

Fig. 32. MSE during model training for ConvDae.

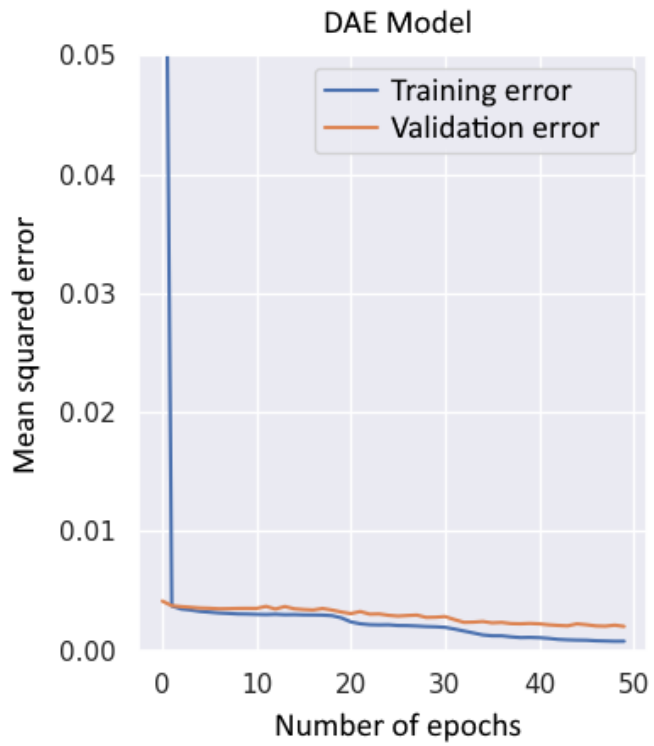

Fig. 33. MSE during model training for DAE.

ments to acquire beam patterns from real devices rather than simulated ones;
- Improvement in the CNN training process, making it independent of invalid values for training;
- Mechanism for inserting new devices into the network so that the model learns new patterns without forgetting the old ones and without necessarily having to relearn all the already recognized patterns.

These improvements can be made in a future project, but they were not implemented due to the impossibility of completion in a timely manner, as there were no real devices available for measurement nor sufficient computational power to generate the databases within the project timeframe.

As a direction for future work, we propose exploring anomaly detection in the frequency domain rather than relying solely on spatial-domain analysis. While this study focused on reconstructing beam signatures derived from angular measurements in MIMO antenna arrays, it is plausible that spectral representations of signals—particularly those based on channel features such as CSI or SRS, could reveal abnormal device behaviors. Transforming temporal or spatial data into the frequency domain using techniques like the Fourier Transform may help highlight hidden interference patterns, irregular modulation, or spectral deviations that are not easily captured in conventional analysis. A comparative study of detection models operating in the time, space, and frequency domains could, therefore, enhance our understanding of the strengths and limitations of each approach within the context of physical-layer security in 5G networks.

VII. CONCLUSION

Communication systems have evolved to provide connections with high key performance indicators (high data rates, low latency, etc.) while also being designed with cybersecurity concerns. However, as attackers continuously evolve their tactics, cybersecurity protection must be constantly improved. In this paper, we use convolutional networks to analyze the beam pattern transmission and, according to the signature, indicate a non-conforming transmission, which can imply the presence of an intruder.

The results demonstrate the feasibility of applying artificial intelligence to the physical layer security of 5G networks, specifically in detecting spoofing attacks based on the manufacturing characteristics of antennas. High precision and specificity were obtained for the three models trained for detection, highlighting the high performance of CNN, which surpassed the other models in accuracy and recall; in general, all performed above 80% in accuracy. The success of model training, even for a simplified scenario, demonstrates the feasibility of implementing such a mechanism in 5G networks. Since the process utilizes one of the phases of beam management, the data can be easily obtained from a gNodeB and sent to the server, where the trained model would be located to recognize the devices.

Future work will focus on implementing the models in more realistic scenarios, considering multipath propagation, channel fading, and other characteristics in open and closed urban

environments. Additionally, the impact of user equipment mobility will be explored, ensuring the robustness of the detection mechanism in dynamic network conditions. Further studies could also investigate adaptability to different antenna types and configurations, as well as real-time processing optimizations to facilitate practical deployment in 5G infrastructures.

VIII. References Section

References

[1] S. Sullivan, A. Brighente, S. A. Kumar, and M. Conti, "5g security challenges and solutions: a review by osi layers," *IEEE Access*, vol. 9, pp. 116294–116314, 2021.

[2] A. Handa, A. Sharma, and S. K. Shukla, "Machine learning in cybersecurity: A review," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 9, no. 4, p. e1306, 2019.

[3] J. Suomalainen, A. Juhola, S. Shahabuddin, A. Mämmelä, and I. Ahmad, "Machine learning threatens 5g security," *IEEE Access*, vol. 8, pp. 190822–190842, 2020.

[4] P. Sharma, S. Jain, S. Gupta, and V. Chamola, "Role of machine learning and deep learning in securing 5g-driven industrial iot applications," *Ad Hoc Networks*, vol. 123, p. 102685, 2021.

[5] A. Afaq, N. Haider, M. Z. Baig, K. S. Khan, M. Imran, and I. Razzak, "Machine learning for 5g security: Architecture, recent advances, and challenges," *Ad Hoc Networks*, vol. 123, p. 102667, 2021.

[6] M. S. Almutairi, "Deep learning-based solutions for 5g network and 5g-enabled internet of vehicles: advances, meta-data analysis, and future direction," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–27, 2022.

[7] I. Sahni and A. Kaur, "A systematic literature review on 5g security," *arXiv preprint arXiv:2212.03299*, 2022.

[8] A. Mozo, A. Karamchandani, S. Gómez-Canaval, M. Sanz, J. I. Moreno, and A. Pastor, "B5gemini: Ai-driven network digital twin," *Sensors*, vol. 22, no. 11, p. 4106, 2022.

[9] R. W. Heath Jr. and A. Lozano, *Foundations of MIMO Communication*. Cambridge University Press, 2018.

[10] C. A. Balanis, *Antenna theory: analysis and design*. John wiley & sons, 2016.

[11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[12] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into deep learning," *arXiv preprint arXiv:2106.11342*, 2021.

[13] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[14] 3GPP, "Study on New Radio (NR) access technology ," Technical Report (TR) 36.912, 3rd Generation Partnership Project (3GPP), 09 2018. Version 15.0.0.

[15] Y.-N. R. Li, B. Gao, X. Zhang, and K. Huang, "Beam management in millimeter-wave communications for 5g and beyond," *IEEE Access*, vol. 8, pp. 13282–13293, 2020.

IX. Biography Section

João Pedro M. Gomes (Engineer) He holds a bachelor's degree in Electrical Engineering from the Universidade Federal de Campina Grande (2023). Participated in UFCG's LABMET (Laboratório de Metrologia) as student, where he worked with machine learning applied to 5G cybersecurity. Currently is a verification engineer working with ASIC development in HwIT company, with experience in verifying ASICs related to Digital Signal Processing (DSP), Foward Error Correction (FEC) and communication protocols (APB, SPI, AHB, AXI). Mobile networks.

Matheus Vilarim P. dos Santos (Graduate Student Member, IEEE) He holds a Bachelor's degree in Electrical Engineering from the Federal University of Campina Grande (2021) and a postgraduate degree in Information Security. He is a researcher at the Signal and Information Processing Laboratory (LAPSI) at UFCG, with experience in security architecture and 5G regulation theory, working on the UFCG/ANATEL Decentralized Execution Term. He is committed to studying the Cybersecurity of Cyber-physical Systems, more specifically, the National Infrastructure of the Electric Power System. He holds a Master's degree in Electrical Engineering from UFCG, with a concentration in Information Processing, where he applies his studies to methods for locating Jammers in mobile networks. He also worked on the Hackers do Bem Project at RNP, where he developed a simulation framework for teaching cybersecurity, applying virtualization and LLMs to support student learning. He is currently a PhD student in Software Engineering at CESAR School and a researcher at the cybersecurity competence center, CISSA.

Edmar C. Gurjão (Senior Member, IEEE) Graduated in Electrical Engineering from Universidade Federal da Paraíba (1996), master in Electrical Engineering from Universidade Federal da Paraíba (1999) and PhD in Electrical Engineering from Universidade Federal de Campina Grande (2003). Visiting professor at Notre Dame University (USA) in 2012. Actually is professor of Electrical Engineering Department at Universidade Federal de Campina Grande and in the Master Degree Program in Science and Technology in Health at Universidade Estadual da Paraíba. Experience in Electrical Engineering with emphasis in Compressed Sensing, Software Defined Radio and Signal Processing and Cybersecurity. Senior Member of IEEE and member of the Brazilian Society of Telecommunications (SBrT). Co-author of Introduction to Signal and Systems (in Portuguese) 2015, and Digital Signal Processing, Momentum Press, 2018.