Adversarial Robust ViT-based Automatic Modulation Recognition in Practical Deep Learning-based Wireless Systems

Gen Li[1†], Chun-Chih Lin[1†], Xiaonan Zhang[2], Xiaolong Ma[1], Linke Guo[1*]
[1]*Clemson University,* [2]*Florida State University*
Email:{gen, chunchi, xiaolom, linkeg}@clemson.edu, xzhang@cs.fsu.edu

***Abstract*—Advanced wireless communication systems adopt deep learning (DL) approaches to achieve automatic modulation recognition (AMR) for spectrum monitoring and management, especially in the spectrum bands supporting diverse co-existing wireless protocols. In practical wireless environments, wireless signals can easily get compromised by malicious noise, intentional interference, and adversarial attacks, reducing the effectiveness of AMR. By exploiting DL model vulnerabilities, an undetectable perturbation added to the wireless signal can cause misclassification, resutling in serious consequences including decoding errors, throughput degradation, and communication disruption. Facing the limitations of existing works on defending against wireless adversarial attacks, this work innovates the Transformer model to design an adversarial robust AMR driven by exploring temporal correlation in time-sequence wireless signals. Instead of directly applying the Vision Transformer (ViT), we first innovate a feature extraction module specifically for radio frequency (RF) signals from both the time and frequency domains, together with an adaptive positional embedding to the Transformer encoder for enhancing AMR accuracy. To mitigate the noise effect in practical wireless communication, we then propose a noise-adaptive adversarial training scheme on the developed Transformer-based model using adversarial examples crafted by white-box attackers. To show the scheme's efficiency, effectiveness, and robustness, our proposed design has been thoroughly evaluated via a self-collected real-world dataset consisting of over 30 million wireless signal data samples with 21 modulation schemes in both indoor and outdoor scenarios. Our results reach a maximum accuracy of 94.17% in AMR classification and 71.2% under adversarial attacks. Besides, for the first time, we demonstrate the robustness of our design under a real wireless adversarial attack in real-time. Datasets and code available in https://github.com/coulsonlee/Robust-ViT-for-AMR-SP2025.**

Keywords: Adversarial Attack, Transformer, Robustness

1. Introduction

Automatic Modulation Recognition (AMR) is a crucial technology in modern wireless communication systems, playing a key role in efficient spectrum management [1], dynamic resource allocation [2], and enhanced communication reliability [3]. It is particularly important for identifying the modulation of unknown signals, especially in shared spectrum environments, where signals from overlapping protocols (both legacy and non-standard protocols) are received without clear modulation information due to high-level noise. This capability makes AMR essential for ensuring accurate signal decoding. By recognizing the modulation scheme of received signals, AMR allows systems to adapt to varying conditions and maintain optimal performance. Recent years have witness the integration of deep learning (DL) to AMR, which enables models to automatically extract and learn complex features from raw signal data, surpassing traditional methods in adaptability and performance [4], [5], [6], [7], [8], [9], [10]. However, wireless environments are inherently challenging, plagued by interference from devices, multipath fading, and noise, alongside adversarial attacks. These adversarial attacks exploit DL model vulnerabilities [11], [12], [13], [14], leading to incorrect predictions and severe consequences, such as decoding errors, data loss, reduced throughput, and compromised security. Especially in adversarial attacks on AMR systems, a small and undetectable perturbation added as an interference to the communication channel may cause misclassification of modulation schemes and finally introduce a high bit-error-rate at the receiver and communication disruption. Even worse, as spectrum environments become increasingly crowded, the risk of adversarial attacks grows, with malicious co-existing devices exploiting and crafting interference. These issues are particularly critical in both military and civilian contexts, where reliable communication is paramount. Addressing these challenges is imperative for developing robust AMR systems capable of withstanding adversarial conditions, thus ensuring the reliability and security of wireless communication networks.

To defend against adversarial attacks, current research works in the computer vision domain have developed detection and mitigation techniques such as defensive distillation [15], [16], [17], input pre-processing [18], [19], and adversarial training (AT) [20], [21], [22], which have proven effective in improving model robustness against perturbed images. However, these approaches are not directly applicable to time-sequence wireless signals due to fundamental differences between static images and dynamic signal data,

† *Co-first authors, equal technical contribution.*
∗ *Corresponding author*
DISTRIBUTION STATEMENT A. Approved for public release: distribution is unlimited. OPSEC# 8918.

2025 IEEE Symposium on Security and Privacy (SP) | 979-8-3315-2236-0/25/$31.00 ©2025 IEEE | DOI: 10.1109/SP61157.2025.00030

DOI 10.1109/SP61157.2025.00030

showcasing a series of research challenges in designing an adequate adversarial robust defensive scheme. On the one hand, the current Convolutional Neural Network (CNN)-based AMR works [4], [5], [6], [7], [8] achieve only around 50%-60% accuracy in terms of modulation classification, which will further degrade under adversarial attacks. Even with defensive mechanisms, the CNN model used for AMR suffers from poor performance [23], [24]. This is primarily because CNNs lack the ability to capture temporal correlations inherent in sequential data, a crucial aspect for accurately identifying patterns in wireless signals. On the other hand, as one of the most widely used techniques, AT on adversarial examples is deemed to enhance the robustness of the model. However, the interference levels experienced in real communication channels add another layer of complexity, for which the adversarial perturbation cannot be too large (received packets will be directly dropped) or too small (less effective to cause misclassification). Even worse, existing works investigating wireless adversarial attacks [19], [25], [26], [27] mostly use dataset RML2016.10a [28] and RML2018.01a [29] for validating model robustness, both of which are full of impractical assumptions and erroneous wireless data, making their results less convincing in practical scenarios.

As RF signals are typically continuous analog or digital signals with strong temporal correlations, these correlations allow attention mechanisms [30] to capture the interdependencies between different segments of the signal [31], [32], [33], [34], enabling a better understanding of its structure and features. Motivated by [10], [31], [32], [33] that leverage Vision Transformer (ViT)-based models to classify RF signals, their results (although using RML2016.10a and RML2018.01a) exhibit superior performance compared with their CNN counterparts. The reason behind this is that CNNs may overlook temporal correlations, while attention mechanisms dynamically adjust weights to focus on the most relevant parts of the wireless signal. Hence, **we argue that models with attention mechanisms, i.e., Transformers, may outperform CNNs in handling RF signals via better capturing their temporal dynamics**.

In this work, we focus on designing a fundamental security framework targeting at enhancing the robustness of AMR specifically in adversarial wireless environments. We propose a two-step ViT-based framework rather than only focusing on enhancing the robustness at the receiver. As the first step, instead of shaping wireless signals as images for the ViT, we directly use raw signals and their extracted features as input. Along with a novel adaptive feature/positional embedding design, our ViT-based AMR achieves over 90% of accuracy as opposed to 60% in CNNs. Then, we propose noise-adaptive adversarial training on the developed Transformer model using adversarial examples crafted via white-box attackers. Besides performing experiments on self-collected real-world datasets, we conduct practical wireless adversarial attacks in real-world scenarios to evaluate the proposed design. Our main technical contributions are as follows:

1) We propose a novel sliding window-based mechanism to extract both time and frequency domain features of a wireless signal to enrich feature dimensions for the ViT-based model.
2) To fully leverage the extracted features, we design an adaptive positional embedding using both sinusoidal positional encoding and matrix addition based on the linear transformation of the signal.
3) We develop a noise-adaptive adversarial training algorithm to mitigate the practical impact of noise in generating effective adversarial perturbations.
4) As one of the main contributions to the community, we collect a dataset consisting of more than 30M wireless data samples and 21 different modulation schemes in real-world scenarios. Extensive experiments on this dataset demonstrate the effectiveness, efficiency, and robustness of our design.
5) For the first time, we conduct real-world experiments to demonstrate the robustness of our design under practical adversarial attacks.

2. Background and Motivation

2.1. Limitations of Current Dataset

2.1.1. Dataset Description. Two commonly used datasets in AMR and wireless adversarial attacks are RML2016.10a [28] and RML2018.01a [29] (detail shown in Table 8 in Appendix. A.1). Based on our observation, those two datasets, although being widely used for evaluating the DL performance, have **incorrect** data generation process, consist of **erroneous** signals/noises, and render **impractical** wireless propagation models, making them less convincing in their experimental results, e.g., modulation classification [4], [5], [6], [7], [8], [9], advanced adversarial attacks [19], [25], [26], and defensive approach designs [23]. Note that we only show a glimpse of deficiency found in those datasets, where many more error-prone data in the datasets make them even unusable.

2.1.2. Incorrect Signal Generation. There are inappropriate settings when generating datasets in RML2016.10a and RML2018.01a.

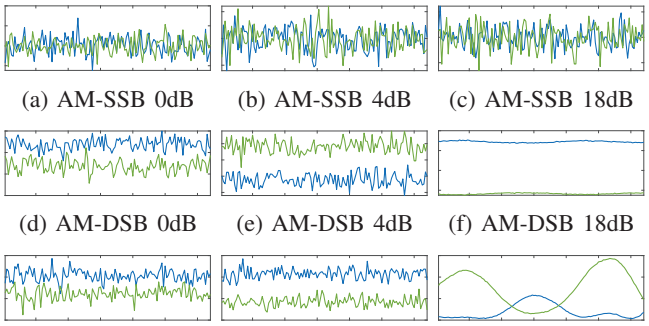
• **Unrealistic Generation Process:** First, all signals used in RML2016.10a are artificially generated with simulated SNRs using GNU radio, which cannot fully capture real wireless environments. Although the signals in RML2018.01a are generated and transmitted using a USRP B210 over the air, the experiment environments are limited to only indoor environments. In practice, wireless signals suffer from channel fading, interference, noise, and the blockage of obstacles, all of which could lead to signal distortion and finally hamper the AMR performance. Only collecting data in a controlled indoor environment is insufficient to evaluate meaningful research designs to be used in real-world applications.

• **Unmatched Frequency Band:** The RML2018.01a dataset consists of signals transmitted over the air on the 900 MHz band, not a common spectrum for signals with the

underlying modulation schemes. In real communication systems, signals transmitted at different frequency bands behave differently regarding transmission ranges, penetration capabilities. More importantly, different noise/interference levels and their impacts, for which this dataset is inappropriate for AMR.

• **Missing Sampling Rates:** While RML2016.10a adopts 8 samples per symbol for generating signals, RML2018.01a does not give any clue about its sampling rate. When transmitting a signal in a real wireless environment, using a higher sampling rate setting leads to generating stretched signals, while a lower sampling rate setting distorts the signal and causes incomplete waveforms.

2.1.3. Erroneous Wireless Signals. Further investigating the dataset in RML2016.10a, we find the modulation schemes provided in RML2016.10a are defective, resulting in false results. For example, numerous samples from AM-DSB and WBFM modulations start from 4 dB to 18 dB SNR containing only noises. We randomly choose samples from the AM-DSB and WBFM modulations list with 0, 4, and 18 dB. Fig. 1 shows the I/Q sample values of the signal data. Unfortunately, Fig. 1d to Fig. 1i demonstrate the value of their I/Q samples are random noises (or even a straight line) as opposed to real signals shown in Fig. 1a to Fig. 1c. Although the authors in [28] believe those random noises may exist at all times and are generated by the silence of the audio (i.e., a single tone), similar waveforms can hardly be found in AM-SSB samples, Fig. 1a to Fig. 1c. We reckon that the single-tone signals should be ruled out since all analog signals would suffer from the issue, and a potent classification mechanism should be able to identify if there are actual signals over the air or if the channel is clean.



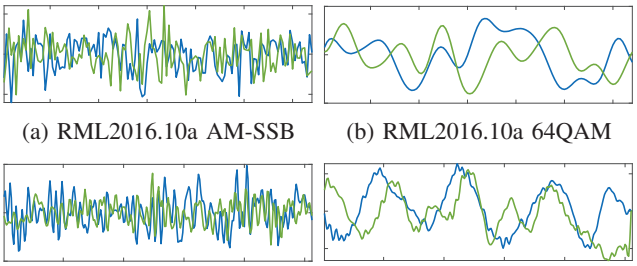
(a) AM-SSB 0dB (b) AM-SSB 4dB (c) AM-SSB 18dB

(d) AM-DSB 0dB (e) AM-DSB 4dB (f) AM-DSB 18dB

(g) WBFM 0dB (h) WBFM 4dB (i) WBFM 18dB

Figure 1: RML2016.10a Dataset Waveform

2.1.4. Impractical Channel Model. The two datasets disregard the practical channel model in generating their signals, for which channel fading, interference, and multi-path effects are all intentionally excluded. In particular, the SNR used in dataset RML2016.10a is generated by synthetic additive white Gaussian noise (AWGN) without actual transmissions in a real wireless environment. Thus, the adversarial attack performance in [19], [25], [26] lacks credibility because the effect of crafted perturbation may deteriorate due to random noises.

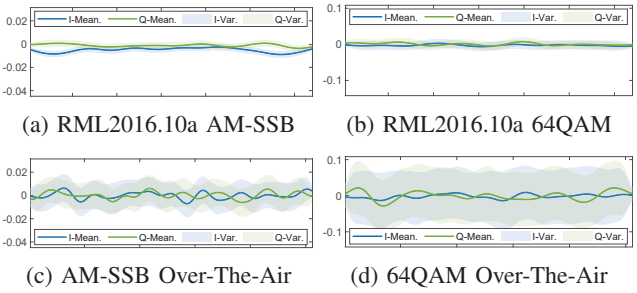


(a) RML2016.10a AM-SSB (b) RML2016.10a 64QAM

(c) AM-SSB Over-The-Air (d) 64QAM Over-The-Air

Figure 2: Sample Signal Comparison

To show the deficiency of the impractical channel model used in the two datasets, we conducted an experiment using USRP X310 to capture signals transmitted in a real wireless environment. As a comparison, two signals (AM-SSB and 64QAM) with 2 dB SNR from RML2016.10a are randomly chosen, as shown in Fig. 2. Although their I/Q samples look similar, their performance after experiencing noisy channels differs significantly, as in Fig. 3. Even with synthetic AWGN in Fig. 3a and Fig. 3b, the simulated signals in RML2016.10a lack variance and fading effect after passing a low-pass filter, compared with real signals transmitted over-the-air as shown in Fig. 3c and Fig. 3d.



(a) RML2016.10a AM-SSB (b) RML2016.10a 64QAM

(c) AM-SSB Over-The-Air (d) 64QAM Over-The-Air

Figure 3: Results over Low-pass Filter. Mean/Variance

In summary, those two datasets are inadequate for evaluating the performance of DL-based AMR under adversarial attack, and to collect a dataset containing authentic signals over-the-air is necessary.

2.2. Design Motivation

2.2.1. Current practices. Despite the prevailing tendency of most AMR frameworks to highlight the exceptional performance of their models using the above-mentioned datasets, practical wireless environments may expose these models to significant vulnerabilities. In Table 1, we demonstrate the significant performance decline on both our proposed framework (detailed in Sec. 4) and CNN-based model [28] when subjected to Projected Gradient Descent (PGD) attacks (a commonly employed white-box attack) across both indoor and outdoor scenarios on our collected dataset. To be specific, the AMR accuracy of the traditional CNN-based approaches has a 80.87% (from 87.57% to 9.67%) and 69.77% (from 78.41% to 11.14%) degradation in the outdoor and indoor scenario, respectively. Meanwhile, the

bit-error rate (BER) at the receiver increases from 3.87% to 49.37% and 3.91% to 40.49% in the outdoor and indoor scenarios, respectively. This level of degradation is nearly equivalent to random guessing, demonstrating the severeness of the adversarial attack to wireless systems. Besides, some statistical modulation recognition methods [35] may offer faster processing speeds but will not work when the adversarial perturbation is added. The reason is that the calculated statistical features, e.g., the mean or variance of Euclidean distance between the received signal and the original signal on constellation, will be distorted under attacks. Hence, the robustness of the AMR model is a crucial issue in practical wireless systems.

TABLE 1: The model performance under attack (we denote a k-step PGD attack as PGD-k).

Dataset	Outdoor (Ours / CNN)	Indoor (Ours / CNN)
Std. Accuracy (%)	92.57 / 87.57	83.60 / 78.41
Atk. PGD-10 (%)	11.70 / 9.67	13.93 / 11.14
Std. BER (%)	2.67 / 3.87	4.25 / 3.91
Atk. PGD-10 BER (%)	39.37 / 49.37	40.50 / 40.49

2.2.2. Design Intuition. Our main objective is to enhance the robustness of AMR under adversarial attacks in practical DL-based wireless systems. To achieve this, the current methodology primarily focuses on exploiting DL approaches such as CNN to capture signal patterns and spatial/temporal dependencies to enhance AMR classification accuracy. Not to mention, the use of orthogonal frequency-division multiplexing (OFDM) in some modulation schemes will scatter the recognizable patterns of basic modulation, making the signals hard to classify using a CNN-based design. As shown in Table 2, the CNN-based design performs significantly worse compared to our designed ViT-based model on both our collected dataset and the RML2016.10a dataset. *The fact is that when the signal experiences adversarial attacks, even with the complicated defensive mechanism implemented, the accuracy will not perform better than the case without adversarial attacks.* Even worse, most defensive approaches cannot adapt to different adversarial attacks (usually specific attack-dependent), further hindering the practicality of AMR used in an adversarial wireless environment.

TABLE 2: The model performance with different model structure. The model's sizes are on a similar scale.

Dataset	CNN-based	Ours ViT-based
Ours (Outdoor)	83.12%	92.57% (↑ 9.45%)
RML2016.10a	58.42%	63.74% (↑ 5.32%)
RML2018.01a	60.19%	64.53% (↑ 4.34%)

Besides generating a new dataset in a real wireless environment, our design methodology is a fundamental shift. Instead of only focusing on a mediocre defensive performance using CNN (50%-60% in [4], [5], [6], [7], [8]), **we propose a two-step method as shown in Fig. 4 to 1) enhance the classification accuracy using a novel ViT-based design, and 2) design a noise-adaptive adversarial training scheme to maintain a high robustness under adversarial attacks**.

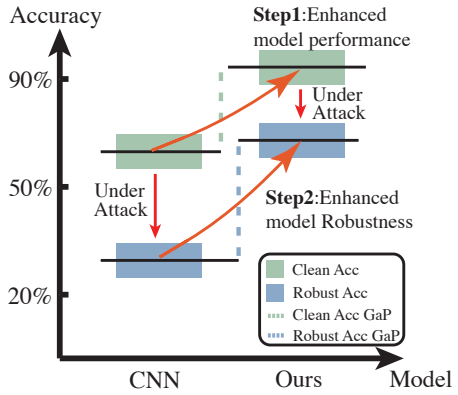


Figure 4: Illustration of our two-step method.

• **ViT-based AMR Design.** Nearly all existing works [4], [5], [6], [7], [8], [9] use CNN as the basic architecture. Originally designed for computer vision tasks, the CNN-based model can only employ local receptive fields (e.g., 3×3 convolutional kernel) to focus on specific regions of the input data, which limits their ability to effectively model long-range dependencies. This limitation can be problematic for time-sequence signals, where long-range dependencies and global interactions are often crucial for AMR. To address this issue, Transformers have emerged as a powerful alternative for processing time-sequence data [10], [31], [32], [33]. Unlike CNNs, Transformers rely on self-attention mechanisms [30] that directly model the interactions between all pairs of input elements, regardless of their positions in the sequence. This global attention allows Transformers to capture long-range dependencies more effectively, making them well-suited for tasks that require modeling complex relationships within wireless signals.

• **Enhancing Adversarial Robustness.** Different from settings in the computer vision domain [20], [21] where perturbation intensity is the primary concern, the main technical challenge of defending against the wireless adversarial attack is that the noise should be considered together with the perturbation generation. Existing works on adversarial attacks [11], [12], [13], [14], [36], [37] and defense [23] either fail to consider the noise effect or use a dataset only containing noise-free signals. Hence, to further enhance the robustness of our ViT-based model, we employ adversarial training by injecting noise-adaptive perturbations into data during training, which is expected to not only mitigate the vulnerability of the model but also exploit the weakness in its decision boundaries.

3. System Overview

We target at a widely-existing wireless communication scenario, where a communication pair is transmitting and receiving wireless signals in an open space. The transmitter sends clean signals to the channel after the modulation. As

the first step of demodulation, the receiver uses a deep-learning-based approach to perform AMR to identify the correct modulation scheme. However, the attacker on the same spectrum band launches the adversarial attack by injecting a crafted signal as a malicious perturbation, resulting in the modulation misclassification and decoding errors.

3.1. Proposed Design

Our proposed robust AMR system is shown in Fig. 5 including the two-step design outlined in Sec. 2.2.2, i.e., ViT-based AMR design and enhancing adversarial robustness. As the first step, using the novel ViT-based models is expected to significantly boost the AMR classification accuracy (on clean samples) by exploiting the long-range dependencies in each signal. To minimize the impact of adversarial attacks on clean signals, our second step is to perform adversarial training using adversarial samples crafted to fool the trained ViT-based model. Instead of generating a wireless signal as a malicious perturbation from an ideal environment, our key innovation is to introduce a noise-adaptive perturbation crafting strategy considering real noises in clean samples. By doing so, the generated practical perturbations included in the adversarial training can further enhance the model robustness in AMR.

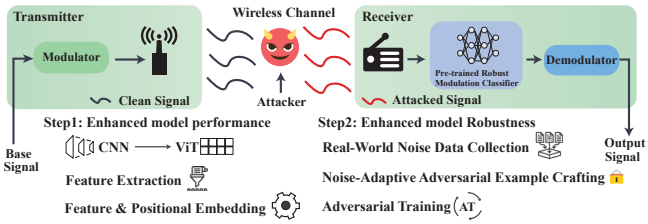


Figure 5: Practical Robust AMC System

3.2. Adversarial Assumptions

In this work, we consider a heterogeneous wireless environment in 2.4GHz, where many wireless protocols co-exist, including Wi-Fi, Bluetooth, ZigBee, unlicensed LTE, LoRa, and radar systems. The crowded spectrum renders opportunities for wireless attackers to use their own dedicated protocol to compromise other's communication by sending crafted interference signals, regardless of protocols and modulation schemes.

• **Adversarial attack.** The adversarial attack introduces subtle perturbations to a received signal to mislead machine learning models, i.e., DL Models in AMR. These perturbations are often imperceptible and comparable to the noise power level but cause signal misclassification. The attack exploits weaknesses in the model's decision-making, aiming to degrade its performance in recognizing modulation types or other signal features.

• **Attacker.** The attacker can be any wireless device that can transmit signals at any location near the victim with a given protocol. The characteristics of the transmitted signal, including transmission power, packet design, and channel selection, follow its dedicated protocol stack, which enables the attacker to craft arbitrary perturbations to launch the adversarial attack on the victim. We assume that the attacker knows the receiver's location and the frequency band used for AMR to receive signals. The attacker is also assumed to be able to detect and measure the surrounding noise [38].

• **Receiver (Victim).** The receiver is assumed to adopt a DL model to classify and recognize the modulation scheme of received signals. Upon receiving the signal, the receiver will let the signal go through the regular de-noising and then get classified. The DL model cannot differentiate whether the received signal has been compromised via adversarial perturbations other than viewing the DL output result.

4. Transfomer-based AMR Design

4.1. Design Overview

Simply applying the Transformer design to the AMR will not have a superior performance, mainly due to the lack of a comprehensive representation of the wireless signals. In practice, each wireless signal will show differently on both time and frequency domain when adopting a specific modulation scheme. Hence, for the first time, as shown in Fig. 6, we renovate the basic Transformer design by integrating the feature extraction on both the time and frequency domains to enrich the feature dimensions in the hope of enhancing the understanding of time-sequence wireless signals. After dividing the signal into non-overlapping patches, our next innovation is to design both feature embedding (for time and frequency domain features) and positional embedding module to learn the relative positions of small patches. Then, we pass the processed signal into the Transformer encoder layers. Finally, the modulation prediction task will be done by a Multi-Layer Perceptron (MLP) head, which takes the output from the Transformer layers and transforms it into a classification output, i.e., each class represents a specific modulation type.

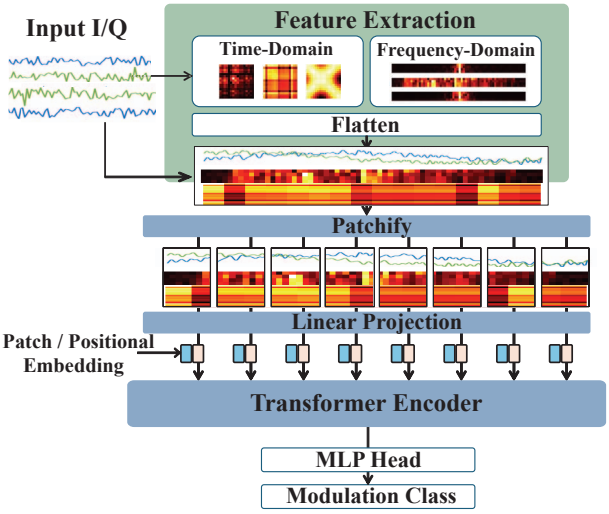


Figure 6: System Overview

4.2. Wireless Signal Feature Extraction

4.2.1. Design Challenges. The Vision Transformers [34] has been used in various computer vision tasks including image classification, object detection, and segmentation, which motivates many works in the area of signal processing [4], [9], [28], [39] to convert wireless signal into images to align with established models designed for visual tasks such as ResNet [40]. Unfortunately, wireless signals inherently possess unique characteristics that distinguish them from natural images. **First**, wireless signals often exhibit a temporal or sequential nature, where the order of this type of sequential data points carries significant meaning. Hence, treating wireless signals as static images may disrupt this inherent temporal structure, potentially resulting in information loss or misinterpretation. **Second**, wireless signals have varying sampling rates and durations, making it challenging to fit them into fixed-size image representations without compromising important details or introducing artifacts. **The last but not the least**, wireless signals often contain complex patterns and dependencies across different scales. These long-range dependencies are crucial for accurate modeling and analysis but may not be effectively captured by models designed primarily for local spatial patterns in natural images.

• **Our Approach.** Real wireless signals are modulated using both time-domain and frequency-domain methods, embedding their features in different ways. To accurately capture these features, we employ two separate feature extraction procedures, which are then concatenated with the raw data and fed into the Transformer.

4.2.2. Time-Domain Feature Extraction. Typically, when signals are modulated in the time domain, a predefined waveform is used repetitively for constructing the transmitted waveform. For example, Bluetooth uses Gaussian Minimum Shift Keying (GMSK) to modulate the signal, and the root-raised cosine (RRC) filter is used to construct the waveform. These signals inherently have a high auto-correlation in the time domain. While different protocols use different signal filters to generate the signal, different patterns can be extracted. In particular, the received signal can be expressed as a time series vector as,

$$x[t] = [x_1, x_2, \ldots, x_L], \tag{1}$$

where L is the total length of the recorded signal. We use a sliding window to choose the section to extract the feature.

$$x_s[t] = [x_i, x_{i+1}, \ldots, x_{i+N-1}], \tag{2}$$

where $1 \leq i$ and $i + N - 1 \leq L$. We then use another sub-sliding window to get a portion of the array x_s and use each of the sub-array to create a matrix $\mathbf{F_T}$ as follows,

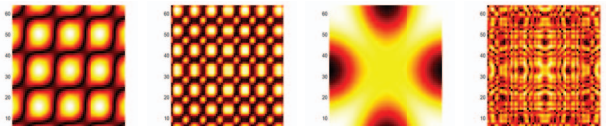
$$\mathbf{F_T} = \begin{bmatrix} x_s[1], & x_s[2], & \ldots, & x_s[M] \\ x_s[2], & x_s[3], & \ldots, & x_s[M+1] \\ \vdots & \vdots & \ddots & \vdots \\ x_s[M], & x_s[M+1], & \ldots, & x_s[N] \end{bmatrix}. \tag{3}$$

The M is the length of the sub-sliding window, and each row in $\mathbf{F_T}$ represents a slice of the input signal. To calculate the auto and cross-correlation, we have,

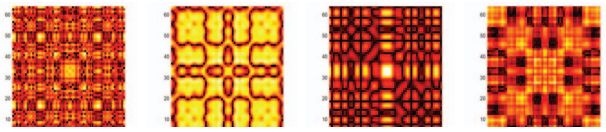
$$\mathbf{F_{T,corr}} = \mathbf{F_T} \cdot \mathbf{F_T}^\top. \tag{4}$$

The diagonal elements in $\mathbf{F_{T,corr}}$ represent the auto-correlation of the sliced signals, and others are the cross-correlation to different slices, by which we can extract the time-domain correlation feature.

We evaluate commonly used modulation schemes in wireless communication and demonstrate the extracted time-domain features in Fig. 7. In particular, both AM-SSB and AM-DSB have very similar generating signal processes, yielding that Fig. 7a and Fig. 7b demonstrate a high similarity to each other. Obviously, their features are very different from others such as FSK in Fig. 7c and Fig. 7d. Similarly, Fig. 7e, Fig. 7h, and Fig. 7g show that their time-domain features are alike mainly due to all of them are generated from phase-plus-amplitude modulation process.



(a) AM - DSB (b) AM - SSB (c) FSK (d) CPFSK



(e) BPSK (f) WBFM (g) PAM4 (h) 16QAM

Figure 7: Time-domain Feature

4.2.3. Frequency-Domain Feature Extraction. Unlike time-domain features, many protocols (e.g., Wi-Fi) embed the coded information in the frequency domain and then apply the inverse Fourier Transform (IFT) to generate time-domain signals for transmission. Rather than extracting their time-domain features, we propose to directly extract features embedded in the frequency domain. Hence, we create a discrete Fourier transform (DFT) matrix with the matrix size $M \times M$ to match the sub-slice we had in the previous time-domain feature extraction,

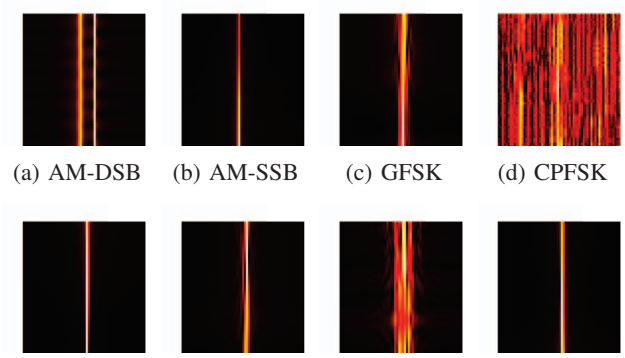
$$\mathbf{DFT} = \frac{1}{\sqrt{N}} \begin{bmatrix} 1, & 1, & \ldots & 1 \\ 1, & \omega, & \ldots & \omega^{N-1} \\ 1, & \omega^2, & \ldots & \omega^{2(N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1, & \omega^{N-1}, & \ldots & \omega^{(N-1)(N-1)} \end{bmatrix}, \tag{5}$$

where $\omega = e^{-2\pi i/N}$. The Fourier transform of all the sub-slices can be obtained by multiplying the DFT matrix with the $\mathbf{F_T}$ as follows,

$$\mathbf{F_F} = \mathbf{DFT} \cdot \mathbf{F_T}^\top. \tag{6}$$

Then, we can observe the changes in the frequency domain in each patch.

As an example, we conduct the frequency-domain feature extraction on 6 modulation schemes in Fig. 8. The most obvious feature is the response to the bandwidth of the modulated signals. For example, the AM-SSB in Fig. 8b shows only a single side of the frequency response compared to AM-DSB as in Fig. 8a. Fig. 8d shows that the CPFSK uses almost the whole bandwidth for modulating the signal.

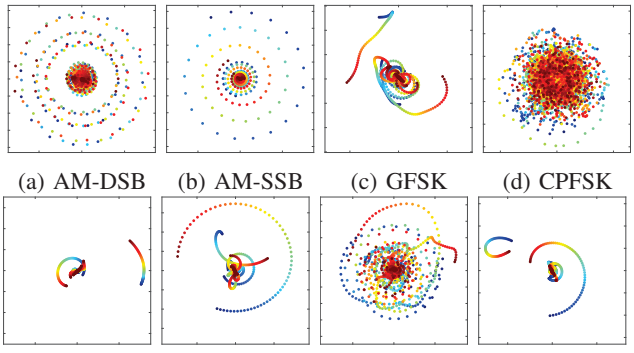


(a) AM-DSB (b) AM-SSB (c) GFSK (d) CPFSK

(e) 8PSK (f) QPSK (g) PAM4 (h) 64QAM

Figure 8: Frequency Feature Throughout Time

• **Joint Time and Frequency-domain Features.** The slicing window design in the time domain helps us observe the transition of the frequency features. The drawn I-Q values as coordinates form the diagrams mimicking the constellation diagram as shown in Fig. 9. The GFSK in Fig. 9c, PAM4 in Fig. 9g, and QPSK in Fig. 9f show that signals move almost continuously in frequency domain while AM-DSB in Fig. 9a, AM-SSB in Fig. 9b, and CPFSK Fig. 9d indicate those modulations move discretely in frequency-domain. When leveraging those features to enrich the feature dimensions, the developed AMR model is expected to benefit a lot in classifying different modulation schemes.



(a) AM-DSB (b) AM-SSB (c) GFSK (d) CPFSK

(e) 8PSK (f) QPSK (g) PAM4 (h) 64QAM

Figure 9: Constellation Diagram

• **Concatenating Time and Frequency-domain features.** To be used in our Transformer-based design, we flatten both features and concatenate them into the time-frequency input for our Transformer model. In particular, the sliding window is with a size of 64, and thus the final time and frequency-domain feature is a 64×64 matrix according to Eq.(1)-(6). We demonstrate a case study on OFDM Wi-Fi in Appendix B, in which the signals are collected in a real wireless environment.

4.3. Feature Embedding Module

4.3.1. Signal Feature Embedding. One of the unique challenges in this work is how to conduct feature embedding in order to convert the input into a model-processable representation. Different from the traditional Transformer design, we add a new channel, i.e., enriched feature channel which combines both time and frequency-domain features, to match the raw signal data size, as shown in Fig. 10.

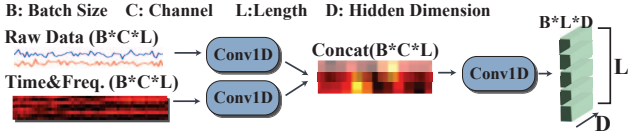


Figure 10: Illustration of Signal Feature Embedding

To effectively process signal data with both enriched and original raw features, we propose a fine-grained information embedding approach. Specifically, we process the enriched feature channels and the raw data channel in parallel using `Conv1D`. Let X_{EN} represent the enriched channel (derived from time and frequency-domain features), and X_R denotes the raw data. The process is defined as follows,

$$Y_{EN} = \texttt{Conv1D}(X_{EN}, \theta_{EN}) \tag{7}$$

$$Y_R = \texttt{Conv1D}(X_R, \theta_R) \tag{8}$$

$$Y = \texttt{Conv1D}(\text{Concat}(Y_{EN}, Y_R, \text{dim} = ch), \theta) \tag{9}$$

where θ_{EN} and θ_R represent the learnable parameters associated with the respective `Conv1D` operations on the enriched feature and raw data channels, respectively. The outputs Y_{EN} and Y_R are then concatenated along the channel dimension (dim). Lastly, a final `Conv1D` operation, parameterized by θ, is applied to the concatenated representation, yielding the composite output Y, which enables the seamless fusion of temporal and spectral information.

• **Discussion.** With the above design, our approach allows the extraction of local information from the signal. The merging design not only preserves the original input size by concatenating the signal along the channel dimension but also dynamically adjusts the merging process according to the characteristics and context of the input data, thereby enhancing performance and adaptability across various signal types. This approach differs from segmenting the signal into tokens or converting it into images for tokenization [7], [8], [9], allowing the Transformer to leverage fine-grained details within the signal to enhance its ability to understand modulation schemes. Moreover, our channel-aware methodology facilitates the aggregation of information from the raw data feature and our extracted time and frequency-domain features. This enables the Transformer model to better utilize both local and global attention mechanisms, thereby improving its overall understanding and interpretation of the signal's characteristics and modulation patterns.

4.3.2. Signal Positional Embedding. The Transformer used in the computer vision domain is usually unaware of the order of the tokens within the sequence at the very

beginning, preventing it from capturing the relative positional information of the data. Hence, positional embeddings are introduced to provide information about the position of tokens within the input sequence. Specifically, ViT [34], the original Transformer [30], and ViT-based AMR [31], [32] adopt absolute positional embeddings (APEs) for encoding tokens, where the data sequence length is usually fixed. However, wireless signal data in AMR scenarios usually has different sequence lengths, making the APEs less effective.

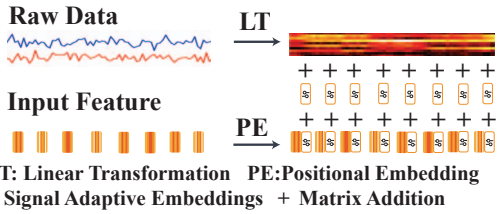


Figure 11: Adaptive Signal Positional Embedding

• **Adaptive Positional Embedding.** Different from APEs, sinusoidal position encoding [30] offers a better generalization capability to different sequence lengths and patterns not seen during training, which allows the model to identify relative positional embeddings by providing unique positional information for each token through sinusoidal functions with varying frequencies. Based on sinusoidal positional encodings, our adaptive design aims to make the positional encodings adaptive to the input signal by learning them jointly with the model parameters, where both adaptive embedding and matrix addition will be used together as shown in Fig. 11. We first adopt the sinusoidal position encoding. For the i-th position in the sequence, the d-dimensional positional embedding vector $\texttt{PE}(pos, 2i)$ and $\texttt{PE}(pos, 2i+1)$ are defined as,

$$\begin{aligned} \texttt{PE}(pos, 2i) &= \sin(pos/10000^{2i/d_{\text{model}}}) \\ \texttt{PE}(pos, 2i+1) &= \cos(pos/10000^{2i/d_{\text{model}}}), \end{aligned} \tag{10}$$

where pos is the position index, and d_{model} is the embedding dimension of the model. By applying different periodic functions (sine and cosine) and dividing by different exponential terms of 10,000, different frequencies of encodings are generated for different positions. For adaptive positional embedding, we apply a linear transformation to the input signal data X,

$$F = W_s X + b_s \tag{11}$$

where the W_s and b_s are learnable matrices and bias vectors. Then, the features F obtained from the linear transformation are added to the positional encoding to find the adaptive position,

$$P_{i_{\text{adaptive}}} = P_i + F, \tag{12}$$

where $P_i \in \mathbb{R}^{d_{model}}$ shows the embedding position.

• **Discussion.** The proposed adaptive positional encodings make the positional embeddings dynamically adjusted based on the input signal. This is achieved by applying a linear transformation to the signal and adding the extracted features to the positional encodings. By learning the positional embedding from the data, the model gains the flexibility to encode complex temporal relationships that may vary across different instances of wireless signals. Please refer to the Appendix C for details on how the attention mechanism learns to recognize different modulation schemes.

5. Enhancing Adversarial Robustness

5.1. Overview

The previous step helps significantly enhance the AMR classification accuracy via the proposed self-attention design, which serves as the foundation for further enhancing its robustness in the adversarial environment. When deep neural networks (DNNs) are used in the AMR system, maliciously crafted small signals sent by the attacker can easily fool the neural network, resulting in high classification errors in AMR [41]. To tackle this issue, existing works [15], [16], [17], [20], [21] spend efforts to establishing robust neural networks, in which the Adversarial Training (AT) methods have demonstrated state-of-the-art robustness. The basic idea of AT is to train a model by exposing it to adversarial examples, which not only mitigates the vulnerabilities of ML models to adversarial attacks but also can exploit weaknesses in the model's decision boundaries. Specifically, the AT aims to solve the following optimization problem,

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{||\delta||_p \leq \epsilon} L(\boldsymbol{\theta}, x+\delta, y) \right] \tag{13}$$

where the input signal sample x and corresponding modulation y follow an underlying data distribution $\mathcal{D}$. In particular, $||.||_p$ is $l_p -$ norm distance metric, and ϵ is a distance constraint set that restricts the perturbation δ to be small. The objective of the maximization is to identify the adversarial perturbation that maximizes the classification loss, i.e., crafting adversarial samples. Meanwhile, the minimization requires the influence on the classification loss brought by adversarial perturbations will be minimized.

As shown in Alg. 1 (in Appendix D), our adversarial training uses adversarial examples (with perturbations) generated for white-box attacks, assuming the model's architecture and parameters are known. In particular, white-box attacks are utilized because they produce stronger and more challenging adversarial examples, simulating a worst-case scenario. Hence, this approach enhances the model's robustness more effectively by rigorously testing its defenses against highly informed attacks.

5.2. Crafting Adversarial Examples

To craft an adversarial example, the imperceptible perturbation δ is added to the clean signal from its distribution. Many existing works have shown the success of misclassification using crafted adversarial examples including fast gradient sign method (FGSM) [11], projected gradient descent (PGD) [20], Carlini-Wagner (C&W) attack [36], which differs in optimization objectives, constraint sets, and computational complexity. While training against non-iterative

attacks such as FGSM can bolster resilience against similar attacks, it may not offer defense against more advanced iterative techniques like PGD attacks. Hence, we choose the PGD attack algorithm to craft adversarial examples, by which the model is exposed to a wide range of potential attack scenarios, leading to enhanced robustness against adversarial perturbations in the input data. Meanwhile, PGD has a solid theoretical foundation [20], making it well-understood and widely studied in the context of adversarial attacks and defenses. Specifically, the formal definition of PGD is,

$$x^{t+1} = \Pi_{x+\boldsymbol{S}} \left(x^t + \alpha \operatorname{sgn} \left(\nabla_x L(\boldsymbol{\theta}, x, y) \right) \right) \tag{14}$$

where t represents the index of iteration, α denotes the step size, and $\operatorname{sgn}(x)$ function returns the sign of a vector. $\Pi_{x+\boldsymbol{S}}$ is the region restriction that bound the adversarial perturbation inside the l_{inf}-ball of radius $\boldsymbol{S}$ centered at the input sample x. In the PGD attack algorithm, the number of iterations K is crucial for both the effectiveness of the attack and the time needed to craft adversarial examples, because each iteration involves a full forward/backward pass to compute the gradient of the loss with respect to the signal.

The PGD attack crafts adversarial samples by iteratively perturbing the input signal to maximize the loss of a neural network in order to result in signal misclassification. Initially, a small random learnable perturbation δ is added to the original input to create a starting point. Then, the perturbation is iteratively updated by computing the gradient of the loss function with respect to the input and taking a step in the direction that increases the loss. This gradient step is constrained by a predefined maximum perturbation ϵ, ensuring that the adversarial sample remains within a small neighborhood of the original input. In our implementation, we maintain the perturbation and the ambient noise at the same power level to ensure the perturbation remains imperceptible as noises. After each gradient step, the perturbed input is clipped to ensure that it remains within the valid input range. The iterative process continues for a predetermined number of steps or until a successful adversarial example is found.

5.3. Noise-Adaptive Adversarial Training in AMR

5.3.1. Design Challenges. The minimization problem in Eq. 13 is the regular training process to learn the representation of adversarial samples to obtain a robust model. The effectiveness of adversarial training in achieving robustness hinges on the potency of the employed adversarial examples, i.e., the perturbation. Hence, the definition of perturbation intensity in wireless signals is crucial for enabling effective adversarial training. Different from image classification, when the perturbation δ on a wireless signal is too large, it can easily be dismissed as excessive noise and can be easily detected. On the other hand, if δ is too small, it may not effectively induce the desired adversarial effects. Therefore, we propose a dynamic strategy as noise-adaptive adversarial training in Fig. 12 to fine-tune the magnitude of the perturbation based on the characteristics of current signals in order to ensure that the perturbation aligns with the order of ambient noise.

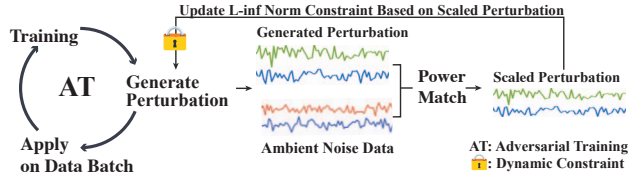


Figure 12: Noise-Adaptive Adversarial Training.

5.3.2. Defining Perturbation Intensity Constraint. Consider a general channel model with AWGN as

$$\mathbf{y} = h * \mathbf{x} + \mathbf{n}, \tag{15}$$

where $\mathbf{y}$ is the received signal, h is the channel response, $\mathbf{x}$ is the transmitted signal, and $\mathbf{n}$ is AWGN with zero mean and variance σ^2. The SNR is defined as the ratio of signal power to noise power as SNR $= \frac{E[|h*x|^2]}{\sigma^2}$, where $E[|h*x|^2]$ is the mean square value (power) of the signal x in terms of the channel response. With this, we further calculate the L_∞ norm of the noise vector n, which is defined as,

$$|n|_\infty = \max_{1 \le i \le k} |n_i| \tag{16}$$

where k is the dimension of the noise vector n. We further prove the SNR is a monotonically decreasing function of $|n|_\infty$ as given in Appendix E. Therefore, given a wireless signal perturbation δ, we aim to dynamically adjust the L_∞ norm of the distance constraint ϵ to limit the perturbation to signal power ($\frac{E[|h*x|^2]}{\delta^2}$) and SNR at the same level.

5.3.3. Adversarial Training Process. The whole adversarial training process is shown in Alg. 1. To generate adversarial samples for training, we need to first determine the wireless signal perturbation δ, which will be added to the input samples to generate $x^{(i)}adv$ in Alg. 1. For a given δ, the distance constraint ϵ is measured by L_∞-norm to ensure that δ falls into a realistic range. Then, for the purpose of best mimicking real-world noises, we aim to match the energy level of the perturbation to that of the noise, by which the impact of the perturbation and noise on the given signal remains the same. With the noise information, the energy level of the perturbation and noise can be directly compared instead of the adjustments in a specific SNR range [23], [42], yielding that the perturbation magnitude can be dynamically tuned. Specifically, the energy of the noise matrix $\boldsymbol{n}$ and perturbation matrix $\boldsymbol{p}'$ can be calculated by summing the squared elements,

$$E_{\mathbf{n}} = \sum_{i,j} n_{ij}^2, \text{and } E_{\mathbf{p}'} = \sum_{i,j} {p'_{ij}}^2 \tag{17}$$

where $n_{ij} \in \mathbf{n}$ and $p'_{ij} \in \mathbf{p}'$. Then, we scale the perturbation matrix $\boldsymbol{p}'$ by a factor of $\sqrt{E_{\mathbf{n}}/E_{\mathbf{p}'}}$ to obtain the final per-

turbation matrix $\boldsymbol{p}$, ensuring its energy matches the energy of the noise matrix $E_{\mathbf{n}}$,

$$\boldsymbol{p} = \sqrt{\frac{E_n}{E_{p'}}}\boldsymbol{p}'. \tag{18}$$

The L_∞-norm of the perturbation matrix $\boldsymbol{p}$, which represents the maximum absolute value of its elements,

$$||\boldsymbol{p}||_\infty = \max_{i,j} |p_{ij}|. \tag{19}$$

By scaling the perturbation matrix $\boldsymbol{p}$ to match the energy of the noise matrix, we ensure that the overall energy level of the perturbation is dynamically adjusted to be comparable to the noise level present in the data. The $||\boldsymbol{p}||_\infty$ will serve as ϵ during adversarial training, as shown in Alg. 1.

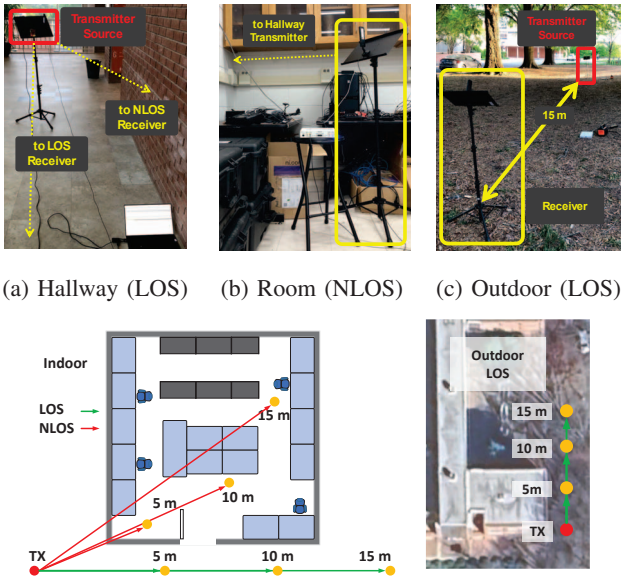
The proposed dynamic strategy helps craft perturbations to adapt the actual noise characteristics without relying on SNR estimations or assumptions about the noise distribution. Based on the observed noise energy, the crafted perturbation used in adversarial training can be more effective to tailor the actual noise conditions. As a result, the model trained from this AT is expected to be more robust in real-world wireless scenarios.

6. Performance Evaluation

6.1. Dataset Collection and Experiment Settings

6.1.1. Wireless Data Collection. We choose the 2.4 GHz ISM band for our data collection and experiments because it supports various coexisting protocols with diverse modulation schemes, making it more accessible for large-scale data collection. Additionally, the coexistence of these protocols better represents the targeted adversarial environment. For example, a ZigBee signal in the 2.4 GHz ISM band can act as a malicious perturbation to Wi-Fi transmissions. We use TI CC2652R1F and GNU Radio to generate signals with different modulation schemes including all the modulations recorded in RML2016.10a plus 802.11 signals modulated in OFDM with BPSK, QPSK, 16QAM and 64QAM, 802.15.4 ZigBee, and 802.15.1 Bluetooth signals with different transmission rate modes listed in Table 9 (refer to Appendix. A.2). Signals generated by GNU Radio are then transmitted via Universal Software Radio Peripheral (USRP) X310 at 2.36 GHz to avoid uncontrollable interference. The details of data collection process and settings are in Appendix A.2.

6.1.2. Environment Settings. We collect more than 30M wireless signal data samples with 21 modulation schemes at different locations with different distances and transmission power levels, in which 3 scenarios have been used as shown in Fig. 13. Modulations without MAC layer information will be recorded as one packet and then dissected into a specific length as an input to the Transformer. On the other hand, modulations with MAC layer information will be recorded as different packets.



(a) Hallway (LOS) (b) Room (NLOS) (c) Outdoor (LOS)

(d) Indoor-Map (e) Outdoor-Map

Figure 13: Dataset Collection Setup

6.1.3. Training Settings. We use 100 training epochs, a batch size of 512, and an initial learning rate of 0.04 for developing the transformer-based model (standard training), in which we employ the SGD optimizer and apply a cosine-annealing learning rate schedule. For the adversarial training, we implement the original PGD-based adversarial training, where the network is trained against an L_∞ adversary with dynamic perturbations δ. Specifically, we choose 10-step PGD for training and 20-step PGD for evaluation, with step sizes α as 0.36 and 0.125, respectively, following the settings in [20], [23]. Besides, we use Auto-Attack [37] and the C&W Attack [36] as comparisons, in which we apply the same setting in the standard training.

6.2. RML Results Comparison

We first compare our proposed framework with current state-of-the-art models using the RadioML2016.10a (RML2016) and RadioML2018.01a (RML2018) datasets to demonstrate the performance gain of our model. While other datasets, such as SPREAD [43] and Sig53 [10], are valuable, they are not specifically designed for modulation classification or lack a variety of established baselines, making it challenging to effectively highlight the superiority of our proposed methods in those contexts. As shown in Table 3, our model achieves the highest test accuracy on both datasets, even with these two erroneous datasets. Although the Top-1 accuracy exhibits marginal improvements compared with MCLDNN [7] (CNN-LSTM-based), SigNet [9] (CNN-based), and CTDNN [31] (Transformer-based), the efficiency gain of our model outperforms them, e.g., 6x fewer parameters and 8x fewer FLOPS than CTDNN, respectively. Among Transformer-based model (FEA-T [44],

TLDNN [45] and CTDNN), we also achieve the highest accuracy. However, it is important to note that those two widely used datasets have significant issues, as discussed in Sec. 2. Many models validated by using these datasets are affected by erroneous wireless signals, resulting in test accuracy that typically hovers around 60% and lacks generalizability to over-the-air signals.

TABLE 3: Comparison of AMR frameworks

Datasets	RadioML2016.10a			RadioML2018.01a		
Method	Parameters	FLOPs	Top-1 Accu.	Parameters	FLOPs	Top-1 Accu.
ResNet [29]	85.52K	3.24M	57.32	164.2K	25.94M	60.91
FEA-T [44]	269.1K	2.19M	55.74	269.0K	19.66M	62.37
MCLDNN [7]	368.7K	41.88M	61.52	370.3K	343.2M	61.92
TLDNN [45]	243.3K	7.89M	62.83	276.7K	22.89M	63.32
SigNet [9]	23520K	-	62.30	-	-	-
CTDNN [31]	2577.2K	331.20M	63.49	-	-	-
Ours	419.0K	42.98M	**63.97**	419.9K	342.43M	**64.53**

As a comparison, we evaluate the test accuracy of the CNN-based VT-CNN2 model [28], a widely used open-source architecture, against our ViT-based design using the self-collected dataset. As shown in Table 4, by using our dataset, even the original CNN-based approach can boost its Top-1 accuracy from 54.17% to 88.64% while our ViT-based design achieves 93.41% accuracy.

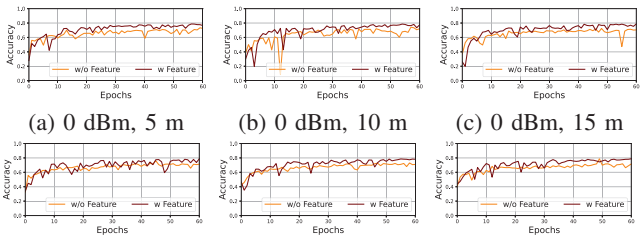
TABLE 4: Test accuracy of our dataset

Model	VT-CNN2(CNN-based)	Ours(ViT-based)
Dataset	Top-1 Accuracy	Top-1 Accuracy
RML2016.10a	54.17	63.97
Our Self-collected	**88.64**	**93.41**

Therefore, the above experimental result further shows the proposed ViT-based model outperforms other models using existing datasets. More importantly, our dataset has shown promising performance gains in both efficiency and accuracy. Moving forward, we will focus on the evaluation only using our self-collected dataset.

6.3. Impact of Feature Extraction

6.3.1. Effectiveness of Feature Extraction. We mainly evaluate the performance with and without the feature extraction module used in the ViT-based model training. We evaluate all 21 modulations collected in our dataset as in Table 9, including both RML2016.10a re-implementation and protocol-driven wireless signals. With a focus on testing accuracy with respect to the training epoch, Fig. 14 shows the performance w/ and w/o feature extraction in an indoor LOS scenario. It is obvious that the training process is much more stable with the feature extraction since the time and frequency domain features help the Transformer learn the definitive characteristics among different modulation schemes. The testing accuracy increases by 5.52%, 3.89%, 6.80%, 3.96%, 9.82%, and 11.62% shown in Fig. 14a to Fig. 14f, respectively, in which both different transmission power levels and transmission distances have been considered.

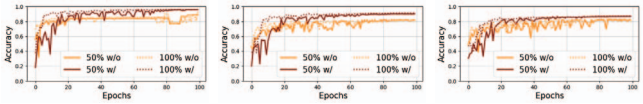


(a) 0 dBm, 5 m (b) 0 dBm, 10 m (c) 0 dBm, 15 m

(d) -10 dBm, 15 m (e) -5 dBm, 15 m (f) 5 dBm, 15 m

Figure 14: Training Process (Indoor LOS)

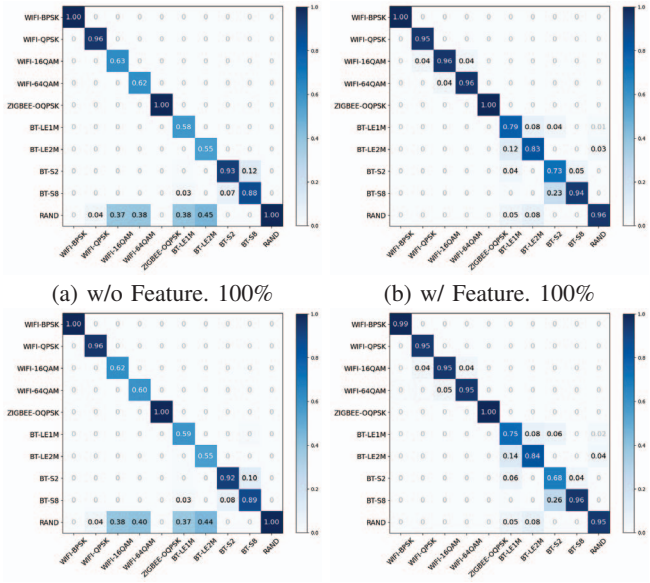
6.3.2. Dataset Size Vs. Accuracy Vs. Efficiency. Since developing the Transformer-based model is data-hungry, it is critical to evaluate the accuracy performance with respect to the training dataset. We specifically investigate the tradeoff between the size of used data samples and model accuracy. As shown in Fig. 15, we demonstrate the training performance w/ and w/o feature extraction using 50% and 100% datasets. In particular, the Transformer (w/ feature extraction) learns slower than the raw I/Q input (w/o feature extraction) at the beginning. Then, the ViT-based model accuracy improves quickly and surpasses the model only using raw I/Q input, yielding a final accuracy improvement of 7.67%, 10.89%, and 6.95% in outdoor LOS, indoor LOS, and indoor NLOS case, respectively.



(a) Outdoor LOS (b) Indoor LOS (c) Indoor NLOS

Figure 15: Dataset Size Vs. Accuracy

To further investigate the reason for accuracy improvement using feature extraction, we show the confusion matrix result in Fig. 16 to demonstrate the accuracy of each class.



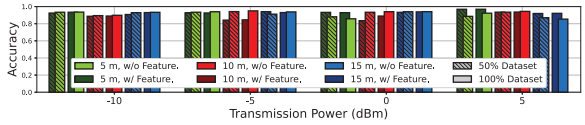
(a) w/o Feature. 100% (b) w/ Feature. 100%

(c) w/o Feature. 50% (d) w/ Feature. 50%

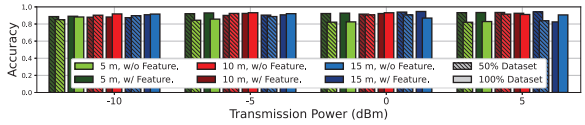
Figure 16: AMR Accuracy (Indoor, LOS, 5m, 0 dBm)

The unprecedented result shows that some modulations (Wi-Fi 16QAM, Wi-Fi 64QAM, BT-LE1M and BT-LE2M) are easily misclassified as noise with either 50% or 100% datasets as in Fig. 16a and Fig. 16c, respectively. With the proposed feature extraction, most Wi-Fi-based modulation schemes have increased their accuracy. The Bluetooth (BT-LE1M and BT-LE2M) using RRC as a waveform generation base also differentiates them from the noise by using time-domain feature extraction.

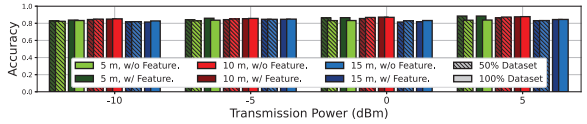
6.3.3. Overall Performance Comparison. Fig. 17 show the overall classification accuracy against the impact of transmission distances, transmission powers, dataset sizes, w/ or w/o features in all three scenarios. With only 50% of training data, the classification accuracy is similar to that using the entire dataset as in Fig. 17. It highlights the efficiency and effectiveness of the proposed framework. In addition to the above discussion, we find out that the increased transmission power further enhances the benefit brought by feature extraction. Compared to previous works, the accuracy obtained from all of our experimental settings is above 80% and close to 98% in the best scenario (outdoor LOS, 5m, 5dBm), showcasing a significant performance gain of the proposed robust AMR.



(a) Outdoor LOS.



(b) Indoor LOS.



(c) Indoor NLOS.

Figure 17: Performance in Different Scenarios

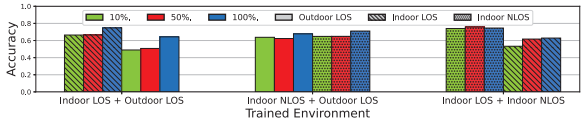
The above results demonstrate that our ViT-based model with the proposed feature extraction module achieves a high classification accuracy with limited data samples. Please refer to the Appendix. F.1 for more results on AMR classification accuracy.

6.4. Cross Dataset Evaluation - Generalization

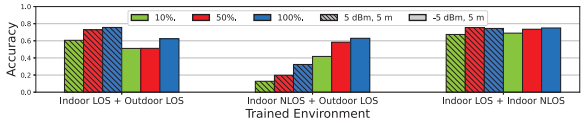
Similar to the Transformer used in NLP fields, for the first time, we will evaluate the generalization capability of our ViT-based model, i.e., recognizing modulation schemes in unseen environments.

6.4.1. Same environment w/ different settings. We first train the Transformer on two selective locations with distances of 5m and 15m and transmission power of 5dBm and -5dBm. Then, we evaluate the Transformer by feeding 10m with 0 dBm at one of the locations (refer to Table 11 in the Appendix for more detail). As shown in Fig. 18a, our model is able to adapt to unseen data with a final accuracy of 64.93%, even outperforming the highest CNN-based AMR accuracy in Table 3. This level of generalization shows our ViT-based model can capture both the characteristics of modulation schemes and the surrounding environment.

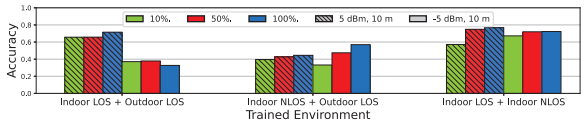
6.4.2. Different environments. We consider the similar setting (same transmission power & distance) and the different setting (different transmission power & different distances). With the setting in Table 12, we show the AMR performance using an unseen dataset with different locations in Fig. 18b and Fig. 18c. Our results show that, on average, with a similar setting, the ViT-based model can achieve 57.59% accuracy, while using the different setting renders the accuracy of 55.29%. The above results further demonstrate that our model trained in a controllable setting is generalizable to different environments with a certain level of accuracy.



(a) Same Environment. Different Setup.



(b) Different Environment. Same Setup.

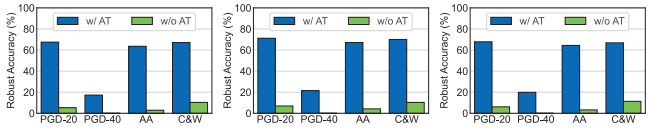


(c) Different Environment. Different Setup.

Figure 18: Cross Dataset Performance

6.5. Robustness Enhancement

With the noise-adaptive AT, we further demonstrate the robustness of our design compared with the one without AT. As shown in Fig. 19, in addition to PGD attacks, we compare our design with more advanced attack algorithms, including AutoAttack (AA) [37] and the Carlini and Wagner (C&W) attack [36]. The robustness accuracy drops significantly with the noise-adaptive AT. When facing attacks like PGD-20 and AA, the test accuracy falls below the level of random guessing, nearly dropping to 0% under the PGD-40 attack. With our proposed AT, the test accuracy improves from a minimum of 21.26% to a maximum of 62.90% under those adversarial attacks.



(a) Outdoor LOS (b) Indoor LOS (c) Indoor NLOS

Figure 19: Different attacks on Models w/ and w/o AT

We also evaluate our proposed AT framework with Safe-AMC [23] under PGD-20 attacks using different architectures as in Table 5. Our method achieves the highest robust accuracy across all different models. Notably, with our proposed ViT-based model, we attain the highest test accuracy of 71.20% and 67.51% on the Indoor LOS and Outdoor LOS datasets, respectively, outperforming Safe-AMC by 4.88% and 5.12%.

TABLE 5: Robust accuracy of different AMR frameworks

Datasets	Indoor LOS		Outdoor LOS	
Method	Ours ViT-based	VT-CNN2 CNN-based	Ours ViT-based	VT-CNN2 CNN-based
Safe-AMC [23]	66.32	54.10	62.39	51.27
Ours	**71.20**	62.18	**67.51**	59.14

6.6. Adversarial Attack Implementation

To further validate the robustness of our design, for the first time, we conduct a real-world experiment in all three wireless environments as shown in Fig. 20.

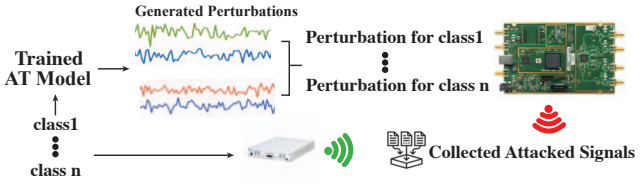


Figure 20: Real-world Implementation

6.6.1. Attacking Process. First, we feed different modulated signals to the well-trained AT model as in Sec. 6.5 and generate the best possible perturbation for attacking the specific signal. Then, we use a USRP B210 to act as an attacker and transmit those perturbations to attack different modulated signals over-the-air. Following the same assumption in Sec. 3.2, the transmission power and the distance between the attacker and the receiver will be chosen meticulously. Fig. 21 shows an example of the antenna placement of the attacker and the receiver, while the received ambient noise w/ and w/o perturbation are also shown on the right of Fig. 21.

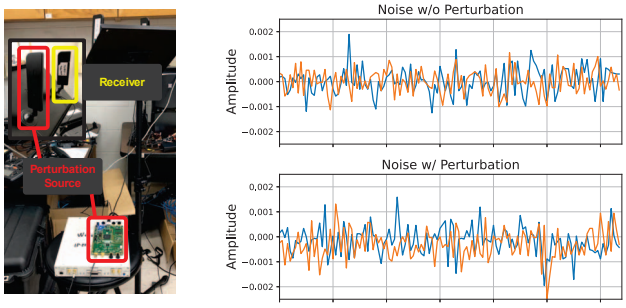


Figure 21: Experiment Environment Setting

6.6.2. Performance Evaluation. Table 6 shows the experiment results in terms of AMR accuracy, including vanilla (no attack), attacked (under adversarial attack), defended (robustness enhanced via AT, and Impl. (real-world implementation) at all locations. The final accuracy achieved 20.97%, 49.37%, and 50.59%. In practice, the Transformer struggles in the outdoor area, because wireless signals experience unexpected environmental changes, such as temperature, winds, humidity, and subtle changes in surrounding objects. The indoor area is a more stable environment, which makes the Transformer achieve over 70% of accuracy performance, even when dealing with unseen attacked signals. Overall, the Transformer outperforms the vulnerable model by 2.95x, 6.11x, and 7.25x.

TABLE 6: Implementation Results

	Vanilla	Attacked	Defended	Impl.
Outdoor LOS	92.57%	5.31%	67.51%	20.97%
Indoor LOS	81.67%	6.94%	71.20%	49.37%
Indoor NLOS	83.59%	6.13%	67.87%	50.59%

6.6.3. Discussion and Analysis. All previous works on AT and testing conducted are in an offline setup, where they first train on a dataset, then craft the corresponding perturbations, and finally inject back to the same dataset before evaluating the performance. In real-world scenarios, however, the developed ViT-based model needs not only to fight against the perturbation generated above but also to recognize the unseen data samples. In our implementation, we first train on a dataset, and then craft the corresponding perturbations based on the actual noise. The perturbation will be directly injected to the transmission over-the-air, thereby generating a new dataset. The defensive performance evaluation will be evaluated based on this new dataset, which is apparently more pertinent to the practical scenario but introduces more challenges. Due to the new channel model, unexpected noises, and new data samples, the practical performance still has room for improvement.

6.7. Training Cost Analysis

6.7.1. Model Efficiency. We finally show the computational cost of inference, standard training, and AT under different DL architectures, in order to evaluate the efficiency of the proposed framework. We specifically evaluate the inference and training cost of our ViT-based model and the CNN-based VT-CNN2 model. As in Table 7, the parameter size in our design is only 14.3% compared with VT-CNN2, underscoring a significant efficiency improvement. For the standard training (ST), our model reduces by 46.5% compared with the VT-CNN2 in terms of FLOPs, while the adversarial training (AT) is down by 45.6%. These two results show a great efficiency gain of our ViT-based design as opposed to the CNN-based model.

TABLE 7: Cost-analysis of model inference and training

Model	Parameters	Inference (×e8)	ST FLOPs (×e16)	AT FLOPs (×e16)
VT-CNN2	2914K	0.42	0.58	3.11
Ours	419K	0.81	0.31	1.69

6.7.2. Real-Time Performance. Given different applications, the real-time requirements vary in terms of latency, jitter, throughput, reliability, etc. For this work, we focus on evaluating the inference delay upon receiving the signal (after the adversarial attack) and the throughput. In terms of throughput, our method processes 1,000 samples in 298 ms, achieving a rate of approximately 3,356 samples per second. With a peak accuracy of 93.41% under normal conditions and 71.20% during adversarial attacks, the effective throughput is approximately 3,135 and 2,383 samples per second, respectively.

7. Related Works

• **Adversarial robustness in ViT.** Several works [46], [47], [48] have found that ViT models are more adversarial robust than CNNs. However, the adversarial technique is mainly designed for CNN-based model. Recently, some works start to design the AT for the ViT. For example, [49] designs a pyramid AT with augmentation techniques to improve the ViT's robustness. [47] designs AT by finding ViTs are more robust to high-frequency adversarial perturbations.
• **Wireless Adversarial Attack.** Previous works [3], [50], [51] show that wireless signals suffers from malicious adversarial attacks, especially in classification. However, those works are all based on the RML2016.10a, which makes the results to be questionable. [52] specifically discuss an OFDM channel under adversarial attack, but their credibility is questionable due the used erroneous dataset.

8. Conclusion

This paper presents a robust AMR system based on Vision Transformer to defend against adversarial attacks in non-cooperative wireless environments. The proposed design leverages a feature extraction module and tailored feature and positional embeddings within the Transformer encoder to enhance model robustness. We further introduce a noise-adaptive AT to mitigate the practical impact of noise in generating effective adversarial perturbation. Through comprehensive real-world experiments based on a self-collected dataset, we validate the efficiency, accuracy, and effectiveness of the proposed framework.

Acknowledgement

We would appreciate the efforts of all anonymous reviewers and the shepherd who helps improve the quality of this paper. The work of L. Guo is partially supported NSF under grant CNS-2008049, CCF-2312616, CCF-2427875, and CNS-2431440. L. Guo's research was sponsored by the Army Research Office and was accomplished under Grant Number W911NF-24-1-0044. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. L. Guo's work was also supported by Clemson University's Virtual Prototyping of Autonomy Enabled Ground Systems (VIPR-GS), under Cooperative Agreement W56HZV-21-2-0001 with the US Army DEVCOM Ground Vehicle Systems Center (GVSC). The work of X. Ma is partially supported by NSF under grant CCF-2427875. The work of X. Zhang is partially supported by NSF under grant CCF-2312617 and CNS-2431439.

References

[1] N. Soltani, K. Sankhe, S. Ioannidis, D. Jaisinghani, and K. Chowdhury, "Spectrum awareness at the edge: Modulation classification using smartphones," in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–10.

[2] B. Jdid, K. Hassan, I. Dayoub, W. H. Lim, and M. Mokayef, "Machine learning based automatic modulation recognition for wireless communications: A comprehensive survey," *IEEE Access*, vol. 9, pp. 57 851–57 873, 2021.

[3] Y. Lin, H. Zhao, X. Ma, Y. Tu, and M. Wang, "Adversarial attacks in modulation recognition with convolutional neural networks," *IEEE Transactions on Reliability*, vol. 70, no. 1, pp. 389–401, 2021.

[4] T. J. O'shea and N. West, "Radio machine learning dataset generation with gnu radio," in *Proceedings of the GNU Radio Conference*, vol. 1, no. 1, 2016.

[5] X. Liu, D. Yang, and A. El Gamal, "Deep neural network architectures for modulation classification," in *2017 51st Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2017, pp. 915–919.

[6] N. E. West and T. O'shea, "Deep architectures for modulation recognition," in *2017 IEEE international symposium on dynamic spectrum access networks (DySPAN)*. IEEE, 2017, pp. 1–6.

[7] J. Xu, C. Luo, G. Parr, and Y. Luo, "A spatiotemporal multi-channel learning framework for automatic modulation recognition," *IEEE Wireless Communications Letters*, vol. 9, no. 10, pp. 1629–1632, 2020.

[8] Y. Zeng, M. Zhang, F. Han, Y. Gong, and J. Zhang, "Spectrum analysis and convolutional neural network for automatic modulation recognition," *IEEE Wireless Communications Letters*, vol. 8, no. 3, pp. 929–932, 2019.

[9] Z. Chen, H. Cui, J. Xiang, K. Qiu, L. Huang, S. Zheng, S. Chen, Q. Xuan, and X. Yang, "Signet: A novel deep learning framework for radio signal classification," *IEEE Transactions on Cognitive Communications and Networking*, vol. 8, no. 2, pp. 529–541, 2021.

[10] L. Boegner, M. Gulati, G. Vanhoy, P. Vallance, B. Comar, S. Kokalj-Filipovic, C. Lennon, and R. D. Miller, "Large scale radio frequency signal classification," *arXiv preprint arXiv:2207.09918*, 2022.

[11] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[12] D. Su, H. Zhang, H. Chen, J. Yi, P.-Y. Chen, and Y. Gao, "Is robustness the cost of accuracy?–a comprehensive study on the robustness of 18 deep image classification models," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 631–648.

[13] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9185–9193.

[14] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," *arXiv preprint arXiv:1611.01236*, 2016.

[15] L. Schmidt, S. Santurkar, D. Tsipras, K. Talwar, and A. Madry, "Adversarially robust generalization requires more data," *Advances in neural information processing systems*, vol. 31, 2018.

[16] P. Nakkiran, "Adversarial robustness may be at odds with simplicity," *arXiv preprint arXiv:1901.00532*, 2019.

[17] T. Chen, S. Liu, S. Chang, Y. Cheng, L. Amini, and Z. Wang, "Adversarial robustness: From self-supervised pre-training to fine-tuning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 699–708.

[18] J. Liu, W. Zhang, Y. Zhang, D. Hou, Y. Liu, H. Zha, and N. Yu, "Detection based defense against adversarial examples from the steganalysis point of view," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4825–4834.

[19] B. Liang, H. Li, M. Su, X. Li, W. Shi, and X. Wang, "Detecting adversarial image examples in deep neural networks with adaptive noise reduction," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 1, pp. 72–85, 2018.

[20] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[21] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan, "Theoretically principled trade-off between robustness and accuracy," in *International conference on machine learning*. PMLR, 2019, pp. 7472–7482.

[22] E. Wong, L. Rice, and J. Z. Kolter, "Fast is better than free: Revisiting adversarial training," *arXiv preprint arXiv:2001.03994*, 2020.

[23] J. Maroto, G. Bovet, and P. Frossard, "Safeamc: Adversarial training for robust modulation classification models," in *2022 30th European Signal Processing Conference (EUSIPCO)*. IEEE, 2022, pp. 1636–1640.

[24] ——, "Maximum likelihood distillation for robust modulation classification," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[25] B. Flowers, R. M. Buehrer, and W. C. Headley, "Evaluating adversarial evasion attacks in the context of wireless communications," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1102–1113, 2019.

[26] B. Kim, Y. E. Sagduyu, K. Davaslioglu, T. Erpek, and S. Ulukus, "Over-the-air adversarial attacks on deep learning based modulation classifier over wireless channels," in *2020 54th Annual Conference on Information Sciences and Systems (CISS)*. IEEE, 2020, pp. 1–6.

[27] D. Adesina, C.-C. Hsieh, Y. E. Sagduyu, and L. Qian, "Adversarial machine learning in wireless communications using rf data: A review," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 77–100, 2022.

[28] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," 2016.

[29] 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*, vol. 12, no. 1, pp. 168–179, 2018.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[31] W. Kong, Q. Yang, X. Jiao, Y. Niu, and G. Ji, "A transformer-based ctdnn structure for automatic modulation recognition," in *2021 7th International Conference on Computer and Communications (ICCC)*. IEEE, 2021, pp. 159–163.

[32] L. Li, C. Qin, G. Li, S. Hu, Y. Xie, and Z. Lei, "Transformer-based radio modulation mode recognition," in *Journal of Physics: Conference Series*, vol. 2384, no. 1. IOP Publishing, 2022, p. 012017.

[33] T. Ya, L. Yun, Z. Haoran, J. Zhang, W. Yu, G. Guan, and M. Shiwen, "Large-scale real-world radio signal recognition with deep learning," *Chinese Journal of Aeronautics*, vol. 35, no. 9, pp. 35–48, 2022.

[34] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[35] T. Huynh-The, Q.-V. Pham, T.-V. Nguyen, T. T. Nguyen, R. Ruby, M. Zeng, and D.-S. Kim, "Automatic modulation classification: A deep architecture survey," *IEEE Access*, vol. 9, pp. 142 950–142 971, 2021.

[36] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*. Ieee, 2017, pp. 39–57.

[37] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in *International conference on machine learning*. PMLR, 2020, pp. 2206–2216.

[38] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys Tutorials*, vol. 11, no. 1, pp. 116–130, 2009.

[39] J. Cai, F. Gan, X. Cao, and W. Liu, "Signal modulation classification based on the transformer network," *IEEE Transactions on Cognitive Communications and Networking*, vol. 8, no. 3, pp. 1348–1357, 2022.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[41] M. Sadeghi and E. G. Larsson, "Adversarial attacks on deep-learning based radio signal classification," *IEEE Wireless Communications Letters*, vol. 8, no. 1, pp. 213–216, 2019.

[42] Y. Lin, H. Zhao, X. Ma, Y. Tu, and M. Wang, "Adversarial attacks in modulation recognition with convolutional neural networks," *IEEE Transactions on Reliability*, vol. 70, no. 1, pp. 389–401, 2020.

[43] H. N. Nguyen, M. Vomvas, T. Vo-Huu, and G. Noubir, "Wideband, real-time spectro-temporal rf identification," in *Proceedings of the 19th ACM International Symposium on Mobility Management and Wireless Access*, 2021, pp. 77–86.

[44] Y. Chen, B. Dong, C. Liu, W. Xiong, and S. Li, "Abandon locality: Frame-wise embedding aided transformer for automatic modulation recognition," *IEEE Communications Letters*, vol. 27, no. 1, pp. 327–331, 2022.

[45] Y. Qu, Z. Lu, R. Zeng, J. Wang, and J. Wang, "Enhancing automatic modulation recognition through robust global feature extraction," *arXiv preprint arXiv:2401.01056*, 2024.

[46] Y. Bai, J. Mei, A. L. Yuille, and C. Xie, "Are transformers more robust than cnns?" *Advances in neural information processing systems*, vol. 34, pp. 26 831–26 843, 2021.

[47] R. Shao, Z. Shi, J. Yi, P.-Y. Chen, and C.-J. Hsieh, "On the adversarial robustness of vision transformers," *arXiv preprint arXiv:2103.15670*, 2021.

[48] S. Paul and P.-Y. Chen, "Vision transformers are robust learners," in *Proceedings of the AAAI conference on Artificial Intelligence*, vol. 36, no. 2, 2022, pp. 2071–2081.

[49] C. Herrmann, K. Sargent, L. Jiang, R. Zabih, H. Chang, C. Liu, D. Krishnan, and D. Sun, "Pyramid adversarial training improves vit performance," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 13 419–13 429.

[50] B. Kim, Y. E. Sagduyu, K. Davaslioglu, T. Erpek, and S. Ulukus, "Channel-aware adversarial attacks against deep learning-based wireless signal classifiers," *IEEE Transactions on Wireless Communications*, vol. 21, no. 6, pp. 3868–3880, 2022.

[51] B. Flowers, R. M. Buehrer, and W. C. Headley, "Evaluating adversarial evasion attacks in the context of wireless communications," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1102–1113, 2020.

[52] A. Bahramali, M. Nasr, A. Houmansadr, D. Goeckel, and D. Towsley, "Robust adversarial attacks against dnn-based wireless communication systems," ser. CCS '21. New York, NY, USA: Association for Computing Machinery, 2021. [Online]. Available: https://doi.org/10.1145/3460120.3484777

[53] "bastibl/gr-ieee802-11," https://github.com/bastibl/gr-ieee802-11.

[54] P. Baldi *et al.*, *Calcolo delle probabilità*. McGraw-Hill, 2011.

TABLE 8: Parameters Setting in [28] and [29]

	RML2016.10a	RML2018.01a
Modulations	BPSK, QPSK, 8PSK, 16QAM, 64QAM, BFSK, CPFSK, PAM4, WB-FM, AM-SSB, AM-DSB	OOK, 4ASK, 8ASK, BPSK, QPSK, OQPSK, 8PSK, 16PSK, 32PSK, 16APSK, 32APSK, 64APSK, 128APSK, 16QAM, 32QAM, 64QAM, 128QAM, 256QAM, AM-SSB-WC, AM-SSB-SC, AM-DSB-WC, AM-DSB-SC, FM, GMSK
Entry per Modulation	1000	4096
Sample Dimension	2×128	1024×2
Generate Methods	Simulated and artificial AWGN	Transmitted Over-the-Air in a Controlled Indoor Environment
Frequency Band	N/A	900 MHz ISM band
SNR Range (in dB)	-20, -18, ..., 10	-20, -18, ..., 30
Sample Rate	Roughly 8 samples per symbol	N/A
Total number of samples	$220,000$	$2,555,904$

Appendix A.
Details of Used Datasets

A.1. Previous Dataset

Table 8 shows the parameters details of RML2016.10a and RML2018.01a Datasets.

A.2. Dataset Collection Details

Table 9 shows the details of our collected dataset.

• **Transmitter.** We modulate wireless signals according to the official standard in Wi-Fi, Bluetooth, and ZigBee protocol. The generated signals only contain the data fields, where no preamble/header or Modulation and Coding Scheme (MCS) information is included.

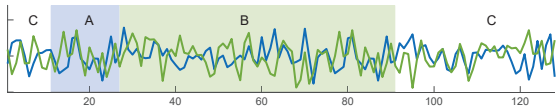
• **Receiver.** The sampling rate is set for 20 MHz for all protocols, and the central frequency is 2,360MHz as in the standard. Taking Wi-Fi as an example, the signal is generated by GNURadio [53] and transmitted by a USRP. Then, we trim off the preamble and header of each signal and then transmit it over-the-air. ZigBee and Bluetooth signals are generated directly from the TI CC2652R1F IoT board controlled by SmartRF Studio 7, with the central frequency and sampling rate strictly following each protocol. Also, we set the transmission in the continuous mode which only generates random messages in data fields.

TABLE 9: Dataset Collection Parameters

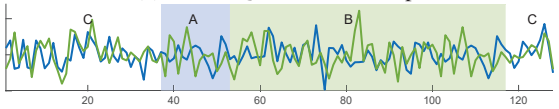
Parameter	Value
Scenario	Outdoor-LOS Indoor-LOS Indoor-NLOS
Frequency Band (MHz)	2,360
Sample Rate (MHz)	0.2 (RML2016.10a), 20 (protocol)
Transmission Power (dBm)	5, 0, -5, -10
Distance (meter)	5, 10, 15
Modulations (RML2016.10a)	BPSK, QPSK, 8PSK, 16QAM, 64QAM, AM-DSB, AM-SSB, PAM4, CPFSK, GFSK, WBFM, Environment Noise
Modulations (protocol)	OFDM-BPSK, OFDM-QPSK, OFDM-16QAM, OFDM-64QAM, OQPSK, BT-LE1M, BT-LE2M, BT-S2, BT-S8

Appendix B.
Case Study on OFDM Wi-Fi

We carry out a case study on Wi-Fi packets with different modulation schemes to show how to use the feature extraction module. The IEEE 802.11 (Wi-Fi), 802.15.4 (ZigBee), and 802.15.1 (Bluetooth) standards require precise packet detection via detecting short- and long-training fields for demodulation and decoding. To achieve this, the protocol will dissect the entire packet into several OFDM symbols after the starting point is detected. Each OFDM symbol in Wi-Fi consists of 64 samples representing the data portion and 16 samples (copied from the last and appended to the beginning of each symbol) representing the cyclic prefix (CP) used to mitigate the inter-symbol interference (ISI) as shown in Fig. 22. For a better description, we label three regions in each of the signals to represent the starting point of the sliding window, where **Region A** shows the starting point falls into the CP, **Region B** indicates it falls into the OFDM symbol region, and **Region C** denotes out of the current symbol.



(a) Wi-Fi QPSK Time Sample

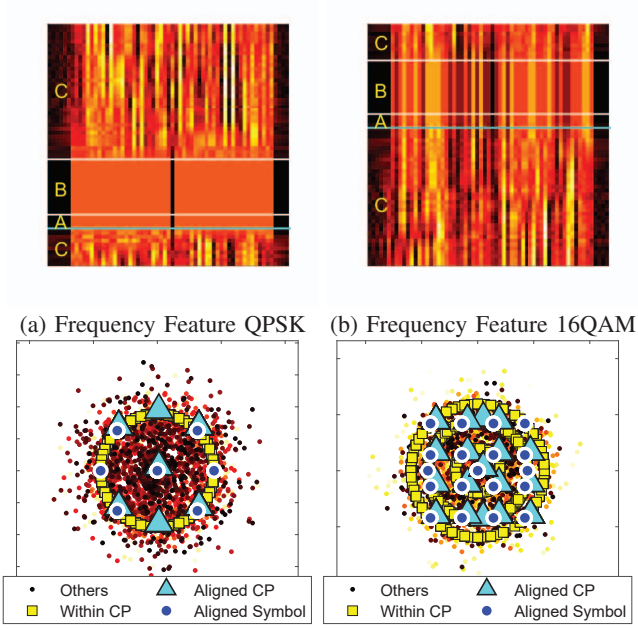


(b) Wi-Fi 16QAM Time Sample

Figure 22: Time-Domain Wi-Fi Signals

When adopting the frequency-domain feature extraction, we observe the frequency features remain unchanged as the sliding window starts in **Region A** and **Region B** as in Fig. 23a and Fig. 23b, respectively. Besides, we capture the I-Q values into the constellation diagram as in Fig. 9. When the sliding window starts at exactly the starting position of CP (beginning of the **Region A**) or the OFDM symbol (beginning of the **Region B**), the constellation diagram exhibits a clear pattern shown as cyan triangle and blue dot in Fig. 23c and Fig. 23d, respectively. Moreover, the I/Q values remain in the same radius to the center (as in yellow squares) if the sliding window is in CP. On the other hand,

I/Q values extracted when the starting point of the sliding window is in **Region C** are scattered in the constellation diagram. Therefore, our frequency-domain feature extraction can provide additional representation of different modulation schemes as long as the proposed sliding window captures the cyclic pattern of a wireless signal.



(a) Frequency Feature QPSK (b) Frequency Feature 16QAM

(c) Constellation QPSK (d) Constellation 16QAM

Figure 23: Wi-Fi Signal Frequency-domain Feature

Appendix C. Attention Maps on Wireless Signals

Attention maps are derived from the model's final layer, utilizing the attention weights from the self-attention mechanism. For each batch sample, we generate a single attention map by averaging the attention weights across all attention heads. These maps provide insight into which parts of the input signal the model emphasizes during decision-making. As shown in Fig. 24, we present attention maps for nine distinct modulation classes. The red boxes highlight the regions (tokens receiving higher attention) that reveal varying patterns across different classes. This indicates that the model is effectively distributing attention to different portions of the input signal depending on the class. These diverse patterns underscore the model's adaptability, demonstrating how the ViT architecture dynamically adjusts attention weights to capture the unique characteristics of each signal type.

Appendix D. Adversarial Training Algorithm

The Alg. 1 shows the process of adversarial training. The perturbation ϵ is dynamically decided by the noise level. Adversarial examples are generated iteratively by applying small perturbations to input data in order to maximize the loss of the model. These perturbed examples are then used to train the model.

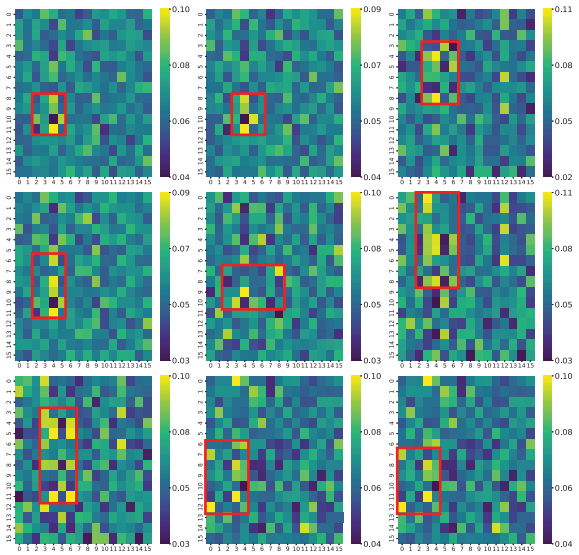


Figure 24: Illustration of last layer attention map.

Algorithm 1: Adversarial Training in AMR

$\mathcal{D}$: training data
θ: initial model parameters
α: learning rate
ϵ: maximum perturbation
τ: Total training iteration
t: current training iteration
while $t < \tau$ **do**
  Sample a minibatch of examples $x^{(i)}, y^{(i)}$ from $\mathcal{D}$
  Generate adversarial examples $x^{(i)}adv$ by solving:
  $x^{(i)}adv = \arg\max_{||x'-x^{(i)}||\infty \leq \epsilon} L(x', y^{(i)}; \theta)$
  Compute loss on adversarial examples:
  $L_{adv} = \frac{1}{m} \sum_{i=1}^{m} L(x^{(i)}adv, y^{(i)}; \theta)$
  Compute gradient: $\nabla\theta L_{adv}$
  Update model parameters: $\theta \leftarrow \theta - \alpha \nabla_\theta L_{adv}$

Appendix E. Proof of SNR as a Decreasing Function

Lemma 1. *A higher SNR implies a smaller* $|n|_\infty$.

Proof. Since n follows a Gaussian distribution, for any given positive number M, we have,

$$P(|n_i| > M) \leq 2 \exp\left(-\frac{M^2}{2\sigma^2}\right). \tag{20}$$

Using the union-bound inequality [54], we obtain,

$$P(|n|_\infty > M) \leq 2k \exp\left(-\frac{M^2}{2\sigma^2}\right) \tag{21}$$

As $M \to \infty$, the right-hand side approaches to zero, indicating $|n|_\infty$ is bounded. This upper bound becomes smaller

when σ is smaller (i.e., when SNR is larger). Therefore, a higher SNR implies a smaller $|n|_\infty$, and vice versa. □

Theorem 1. *SNR is a monotonically decreasing function of* $|n|_\infty$.

Proof. We can use a sufficiently large value of $|n|_\infty$, denoted as M, to estimate σ. Hence, the SNR can be written as $\exp\left(-\frac{M^2}{2\sigma^2}\right) = e$. Solving for σ, we get

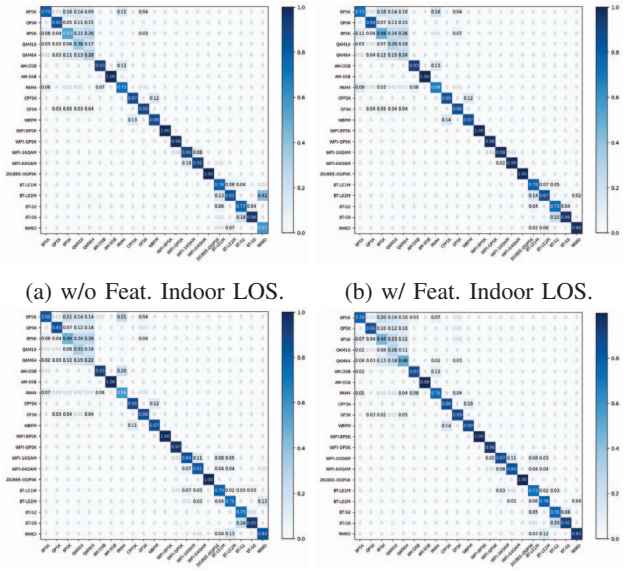
$$\sigma = \frac{M}{\sqrt{2\ln(2k/e)}}. \tag{22}$$

Substituting this into the definition of SNR, we can prove SNR is monotonically decreasing with respect to $|n|_\infty$. □

Appendix F. Additional Experiment Results

F.1. Performance - Dataset Size Vs. Accuracy

We further demonstrate the confusion matrix for all 21 modulation schemes.



(a) w/o Feat. Indoor LOS. (b) w/ Feat. Indoor LOS.

(c) w/o Feat. Indoor NLOS. (d) w/ Feat. Indoor NLOS.

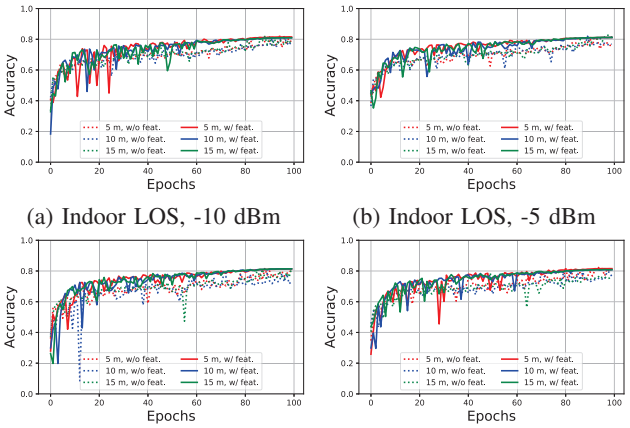
Figure 25: Confusion Matrix - 21 Modulations

F.2. Performance - Distance/Power Vs. Accuracy

Fig. 26 shows the training process with varying transmission power and distances w/ and w/o feature extraction in the Indoor LoS case. Please refer to our Github page for additional results in other scenarios.

F.3. Accuracy Performance

Table 10 lists the additional information for results in Fig. 17. Besides, Table 11 and Table 12 lists the information for cross-dataset accuracy discussed in Sec. 6.4.



(a) Indoor LOS, -10 dBm (b) Indoor LOS, -5 dBm

(c) Indoor LOS, 0 dBm (d) Indoor LOS, 5 dBm

Figure 26: Training Performance, Indoor LoS

TABLE 10: Final Accuracy Comparison

Distance (meters)	**5**	**10**	**15**	
Outdoor LOS	93.88/91.13	88.26/92.98	92.75/91.44	
Indoor LOS	85.78/83.24	85.70/86.23	82.93/83.52	
Indoor NLOS	91.78/84.12	91.05/91.75	90.57/89.25	
Dataset Size	**50%**	**100%**		
Outdoor LOS	91.26/91.57	92.00/92.13		
Indoor LOS	84.42/84.07	85.19/84.59		
Indoor NLOS	91.25/87.59	81.01/89.15		
Transmission Power (dBm)	**5**	**0**	**-5**	**-10**
Outdoor LOS	91.22/92.09	90.16/9.360	90.97/91.51	94.17/90.19
Indoor LOS	83.16/83.41	84.83/84.48	84.83/84.40	86.39/85.04
Indoor NLOS	88.69/89.43	91.41/89.40	92.88/87.65	91.55/87.00

TABLE 11: Cross Dataset Eval. - Similar Environment

Training Dataset		**Location**	**Setting**	**Accu.**
Indoor LOS	(5 m, 15 m)	Indoor LOS	(0 dBm, 10 m)	68.08%
Outdoor LOS	(-5 dBm, 5 dBm)	Outdoor LOS		55.70%
Indoor NLOS	(5 m, 15 m)	Indoor NLOS	(0 dBm, 10 m)	68.22%
Outdoor LOS	(-5 dBm, 5 dBm)	Outdoor LOS		58.11%
Indoor LOS	(5 m, 15 m)	Indoor LOS	(0 dBm, 10 m)	72.37%
Indoor NLOS	(-5 dBm, 5 dBm)	Indoor NLOS		66.08%
Overall Avg.	-	-	-	64.93%

TABLE 12: Cross Dataset Eval. - Different Environment

Training Dataset		**Location**	**Setting**	**Accu.**
Indoor LOS	(5 m)	Indoor NLOS	(5 dBm, 5 m)	69.68%
Outdoor LOS	(-5 dBm, 5 dBm)		(-5 dBm, 5 m)	54.90%
Indoor NLOS	(5 m)	Indoor LOS	(5 dBm, 5 m)	21.63%
Outdoor LOS	(-5 dBm, 5 dBm)		(-5 dBm, 5 m)	54.37%
Indoor LOS	(5 m)	Outdoor LOS	(5 dBm, 5 m)	72.44%
Indoor NLOS	(-5 dBm, 5 dBm)		(-5 dBm, 5 m)	72.51%
Overall Avg.	-	-	-	57.59%
Training Dataset		**Location**	**Setting**	**Accu.**
Indoor LOS	(10 m)	Indoor NLOS	(0 dBm, 5 m)	67.64%
Outdoor LOS	(-5 dBm, 5 dBm)		(-10 dBm, 5 m)	35.89%
Indoor NLOS	(10 m)	Indoor LOS	(0 dBm, 5 m)	42.24%
Outdoor LOS	(-5 dBm, 5 dBm)		(-10 dBm, 5 m)	45.88%
Indoor LOS	(10 m)	Outdoor LOS	(0 dBm, 5 m)	69.60%
Indoor NLOS	(-5 dBm, 5 dBm)		(-10 dBm, 5 m)	70.52%
Overall Avg.	-	-	-	55.29%

A. Meta-Review

The following meta-review was prepared by the program committee for the 2025 IEEE Symposium on Security and Privacy (S&P) as part of the review process as detailed in the call for papers.

A.1. Summary

This paper explores a Vision Transformer (ViT)-based framework to enhance the robustness of Automatic Modulation Recognition (AMR) in wireless systems for classifying different modulations of received signals. The main innovation of the proposed framework is the two-step approach which will first leverage ViT-based model to enhance the AMR accuracy and then adopt the adversarial training (AT) directly on the trained transformer-based model. Compared with existing convolutional neural network (CNN)-based models, the authors have demonstrated their ViT-based model enhances the AMR accuracy in both attacker-free and adversarial environments. The proposed framework innovates both the traditional transformer architecture and AT specifically for wireless signals. Besides, one of the key contributions is the authors are the first to generate a wireless dataset consisting of 30 million data samples with 21 modulations in real outdoor/indoor wireless environments.

A.2. Scientific Contributions

- Provides a New Data Set For Public Use
- Creates a New Tool to Enable Future Science
- Establishes a New Research Direction

A.3. Reasons for Acceptance

1) The proposed method first enhances the AMR accuracy and then enhances the AMR robustness, rather than directly working on the developed model. For all existing works in this field, the AMR accuracy ranges between 50-60%, largely preventing the model performance under adversarial attacks. The proposed ViT-based model achieves more than 90% of AMR accuracy and more than 70% accuracy under adversarial attacks, even higher than existing models without adversarial attacks. These results from real datasets validate the feasibility and high accuracy of the proposed design.
2) The authors conduct extensive experiments to validate the enhanced robustness of the proposed ViT-based design. With their newly collected datasets, the accuracy performances under both clean and adversarial environments exhibit advanced performance. Besides the prominent performance within the same dataset (used in both training and testing), their generalization performance in Section 6.4 also achieves a high accuracy when performing cross-dataset evaluation.