Frequency-based Automated Modulation Classification in the Presence of Adversaries

Rajeev Sahay, Christopher G. Brinton, and David J. Love
School of Electrical and Computer Engineering, Purdue University
{sahayr,cgb,djlove}@purdue.edu

***Abstract*—Automatic modulation classification (AMC) aims to improve the efficiency of crowded radio spectrums by automatically predicting the modulation constellation of wireless RF signals. Recent work has demonstrated the ability of deep learning to achieve robust AMC performance using raw in-phase and quadrature (IQ) time samples. Yet, deep learning models are highly susceptible to adversarial interference, which cause intelligent prediction models to misclassify received samples with high confidence. Furthermore, adversarial interference is often *transferable*, allowing an adversary to attack multiple deep learning models with a single perturbation crafted for a particular classification network. In this work, we present a novel receiver architecture consisting of deep learning models capable of withstanding transferable adversarial interference. Specifically, we show that adversarial attacks crafted to fool models trained on time-domain features are not easily transferable to models trained using frequency-domain features. In this capacity, we demonstrate classification performance improvements greater than 30% on recurrent neural networks (RNNs) and greater than 50% on convolutional neural networks (CNNs). We further demonstrate our frequency feature-based classification models to achieve accuracies greater than 99% in the absence of attacks.**

***Index Terms*—Adversarial attacks, automatic modulation classification, machine learning, privacy, security**

I. Introduction

THE recent exponential growth of wireless traffic has resulted in a crowded radio spectrum, which, among other factors, has contributed to reduced mobile efficiency. With the number of devices requiring wireless resources projected to continue increasing, this inefficiency is expected to present large-scale challenges in wireless communications. Automatic modulation classification (AMC), which is a part of cognitive radio technologies, aims to alleviate the inefficiency induced in shared spectrum environments by dynamically extracting meaningful information from massive streams of wireless data. Traditional AMC methods are based on maximum-likelihood (ML) approaches [1], which consist of deriving statistical decision boundaries using hand-crafted features to discern various modulation constellations. More recently, deep learning (DL) has become a popular alternative to ML methods for AMC, since it does not require manual feature engineering to attain robust classification performance [2].

Despite their robust AMC performance, however, deep learning models are highly susceptible to adversarial attacks [3], which introduce additive wireless interference into transmitted RF signals to induce high-confidence misclassifications on well-trained deep learning models. In addition to degrading the classification performance of a particular targeted model, adversarial attacks are also transferable to other classification networks that are trained to perform the same task as the targeted classifier [4]. As a result, an adversary can degrade the performance of several deep learning models simultaneously, thus reducing spectrum efficiency and compromising secure communication channels.

In this work, we develop a novel AMC method that is capable of mitigating the effects of transferable adversarial attacks. Specifically, our method learns on frequency domain-based features, as opposed to in-phase and quadrature (IQ) time-domain features, which are traditionally used for deep learning-based AMC. After quantifying the model's performance in the absence of adversarial interference, we consider a wireless channel compromised by an adversary aiming to induce an erroneous modulation constellation prediction at the receiver by injecting interference into the transmitted signal. Although the interference degrades the classification performance of the model trained on IQ features, the frequency feature-based model significantly increases the probability of correctly classifying the perturbed signal, thus mitigating the effects of transferable adversarial interference.

Related Work: The susceptibility of deep learning-based AMC models to adversarial attacks has been demonstrated in prior work [5]–[7]. Such attacks have been found to be more efficient than traditional jamming attacks applied in communication networks [8] and, as a result, present challenges for deep learning deployment in autonomous wireless channels [9]. Yet, limited work has been conducted in exploring the degree to which AMC models are susceptible to interference. Few defenses have been proposed to mitigate the effects of wireless adversarial interference [10], and to the best of our knowledge, no work has explored the extent to which adversarial attacks are transferable between domains (although various domains for classification have been investigated [11]). On the other hand, several defenses have been proposed for defending deep learning image classifiers from adversarial attacks, with no method generally accepted as a robust solution [12]. Nonetheless, even considering the adoption of image classification defenses for AMC is difficult due to the differing constraints placed on the adversary in both settings (e.g., transmit power budget, SNR degradation, visual perceptibly,

This project was supported in part by the Naval Surface Warfare Center Crane Division and in part by the National Science Foundation (NSF) under grants CNS1642982 and CCF1816013.

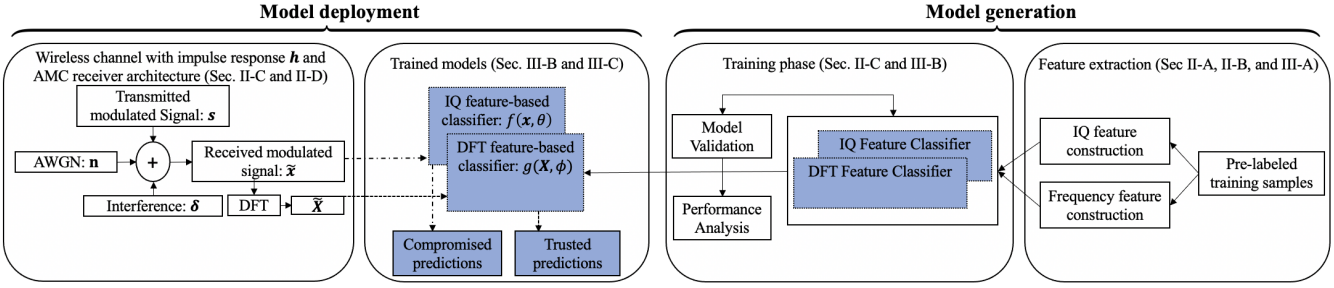

Fig. 1: Our AMC system model with adversarial interference. The shaded blocks correspond to our time and frequency-domain classifiers.

etc.). In this work, we address this challenge by proposing a novel AMC methodology, which allows us to quantify the extent of adversarial transferability in a wireless channel with real-world communication constraints.

Summary of Contributions: The main contributions of this work are as follows:

1) **A novel signal receiver architecture for AMC** (Sec. II-B, II-C, and III-B): We model and develop a robust AMC module consisting of both frequency-based and IQ-based deep learning architectures.
2) **Resilience to time domain adversaries** (Sec. II-D and III-C): We demonstrate that, although an adversary may be able to degrade the classification performance on a time-domain model, their attacks are not well-transferable to our models trained using frequency features.
3) **Best architecture to offset adversary** (Sec. III-C): Our results show that, out of several deep learning architechures, convolutional neural networks (CNNs) have the fastest training times and mitigate the classifier degradation to the greatest extent.

II. Our AMC Methodology

In this section, we outline the wireless channel we consider for AMC as well as the assumptions about the knowledge level of the transmitter, receiver, and adversary. We describe two ways we represent the received signal (Sec. II-A and II-B) followed by the machine learning models we employ for AMC (Sec. II-C). Finally, we describe perturbation methods performed by the adversary to induce misclassifications on the trained models (Sec. II-D). Our overall AMC system model is shown in Fig. 1.

A. Signal Modeling

We consider a wireless channel consisting of a transmitter, which is aiming to send a modulated signal, and a receiver, whose objective is to perform AMC on the obtained waveform and realize its modulation constellation. Specifically, at the transmitter, we consider an underlying data source, $\mathbf{s} = [s[0], \ldots, s[\ell-1]]$, which is modulated using one of C modulation constellations chosen from a set, $\mathcal{S}$, of possible modulation schemes, with each scheme having equal probability of selection. $s[k]$ is the (scalar) value wirelessly transmitted at time k. At the receiver, the collected waveform at each time instance is modeled by

$$x[k] = \sqrt{\rho}(\mathbf{s} * \mathbf{h})[k] + n[k], \tag{1}$$

$n[k]$ represents complex additive white Gaussian noise (AWGN) at time k distributed as $\mathcal{CN}(0,1)$, $\sqrt{\rho}$ denotes the SNR (known at the receiver), $*$ denotes convolution, and $\mathbf{h}$ captures the wireless channel's impulse response. $\mathbf{h}$ also includes radio imperfections such as sample rate offset (SRO), center frequency offset (CFO), and selective fading, none of which are known to the receiver. Furthermore, we assume that the receiver has no knowledge about the channel model or the distribution of $\mathbf{h}$. This general setting motivates an AMC solution using a data-driven approach, as presented in this work, in which the true modulation constellation of the received signal is estimated from a model trained on a collection of pre-existing labeled signals.

B. Domain Transform

At the receiver, we model $\mathbf{x} = [x[0], \ldots, x[\ell-1]]$ using its frequency components obtained from the discrete Fourier transform (DFT). Specifically, the p^{th} component of the DFT of $\mathbf{x}$ is given by

$$X[p] = \sum_{k=0}^{\ell-1} x[k] e^{-\frac{j2\pi}{\ell} pk}, \tag{2}$$

where $\mathbf{X} = [X[0], \ldots, X[\ell-1]]^T$ contains all frequency components of $\mathbf{x}$. We are interested in comparing the efficacy of AMC learning based on $\mathbf{x}$ and $\mathbf{X}$ as feature representations of the input signal. Although both signal representations are complex (i.e., $\mathbf{x}, \mathbf{X} \in \mathbb{C}^{\ell}$), we represent all signals as two-dimensional reals, using the real and imaginary components for the first and second dimension, respectively, in order to utilize all signal components during classification. Thus, we represent all time and frequency domain features as real-valued matrices $\mathbf{x}, \mathbf{X} \in \mathbb{R}^{\ell \times 2}$.

C. Deep Learning Architectures

In this work, we consider the effectiveness of different deep learning architectures for AMC under IQ and frequency features as model inputs. In general, we denote a trained deep learning classifier, parameterized by θ, as $f(\cdot, \theta) : \mathbb{R}^{\ell \times 2} \rightarrow \mathbb{R}^{C}$. This calculates the likelihood, $\hat{\mathbf{y}}$, of an input signal

consisting of IQ features, $\mathbf{x}$, belonging to each of the C modulation constellations. From $\hat{\mathbf{y}}$, the predicted modulation constellation is given by $\arg\max_{i=1,\ldots,C} \hat{y}_i$. Similarly, we denote a deep learning classifier trained using the DFT of the input signal, $\mathbf{X}$, parameterized by ϕ, as $g(\cdot,\phi) : \mathbb{R}^{\ell\times 2} \rightarrow \mathbb{R}^C$, which is trained to perform the same classification task as $f(\cdot,\theta)$ but using the frequency features of $\mathbf{x}$ to comprise the input signal. We analyze the classification performance using the aforementioned signal representations on four common AMC deep learning architectures: the fully connected neural network (FCNN), the convolutional neural network (CNN), the recurrent neural network (RNN) and the convolutional recurrent neural network (CRNN). Each architecture consists of a set of layers and a set of neurons per layer (referred to as units). The specific differences of layer interactions in each considered model are described below. For each model, we apply the ReLU non linearity in its hidden layers, given by $\sigma(a) = \max\{0, a\}$, and a C-unit softmax output layer given by

$$\sigma(\mathbf{a})_i = \frac{e^{a_i}}{\sum_{j=1}^{C} e^{a_j}}, \tag{3}$$

where $i = 1,\ldots,C$ for input vector $\mathbf{a}$. This normalization allows a probabilistic interpretation of the model's output predictions.

FCNN: Our FCNN consists of three hidden layers with 256, 128, and 128 units, respectively. The output of a single unit, u, is given by

$$\sigma\big(\sum_i w_i^{(u)} \cdot a_i + b\big), \tag{4}$$

where $\sigma(\cdot)$ is the activation function, $\mathbf{w} = [w_1,\ldots,w_n]$ is the weight vector for unit u estimated from the training data, $\mathbf{a} = [a_1,\ldots,a_n]$ is the vector containing the outputs from the previous layer (or the model inputs in the first layer), and b is a threshold bias. Each hidden layer applies a 20% dropout rate during training.

CNN: The CNN is comprised of two convolutional layers consisting of 256 and 64 feature maps (each with 20% dropout), respectively, followed by a 128-unit fully connected layer. The output of each feature map in the convolutional layer is given by

$$\sigma\big(\mathbf{v} * \mathbf{a} + b\big), \tag{5}$$

where $\mathbf{v}$ is the filter kernel whose parameters are estimated during training, and $\mathbf{a}$ is the output from the preceding layer. Our model uses a 2×5 and 1×3 kernel for the first and second convolutional layers, respectively.

RNN: The RNN is comprised of a 75-unit long-short-term-memory (LSTM) [13] layer followed by a 128-unit ReLU fully connected layer. Each LSTM unit implements three gates for learning. *Input gates* prevent irrelevant features from entering the recurrent layer while *forget gates* eliminate irrelevant features altogether. *Output gates* produce the LSTM layer output, which is inputted into the subsequent network layer. The gates are used to recursively calculate the internal state of the cell, denoted by $\mathbf{z}_c^{(t)}$ at time t for cell c, at a specific recursive iteration, called a time instance, which is then used to calculate the cell output given by

$$\mathbf{q}^{(t)} = \tanh(\mathbf{z}_c^{(t)})\sigma(\mathbf{p}^{(t)}), \tag{6}$$

where $\mathbf{p}^{(t)}$ is the parameter obtained from the output gate and $\sigma(\cdot)$ is the logistic sigmoid function given by $\sigma(p_i^t) = 1/(1+e^{-p_i^t})$ for the i^{th} element in $\mathbf{p}^{(\mathbf{t})}$.

CRNN: Lastly, we consider a CRNN comprised of two convolutional layers (containing 128 and 64 feature maps with 2×5 and 1×3 kernels, respectively) followed by a 32-unit LSTM layer.

Unless otherwise noted, each model is trained using the Adam optimizer [14], 75 epochs, a batch size of 64, and the categorical cross entropy loss function given at the output by

$$\mathcal{L}_n = \sum_{j=1}^{C} y_j \log(\hat{y}_j) \tag{7}$$

for each sample n and

$$\mathcal{L} = -\frac{1}{N}\sum_{n=1}^{N} \mathcal{L}_i, \tag{8}$$

over the entire training set $n = 1,\ldots,N$, where $y_j = 1$ if the ground truth label of the sample is modulation class j and $y_j = 0$ otherwise.

D. Adversarial Interference

In addition to the transmitter and receiver, our considered communication network also consists of an adversary, whose objective is to induce a misclassification on the trained AMC model. The adversary will perturb the received signal by injecting wireless interference, which we will denote $\boldsymbol{\delta} : \boldsymbol{\delta} \in \mathbb{R}^{\ell\times 2}$, into $\mathbf{x}$ during transmission. For a given design of $\boldsymbol{\delta}$, the resulting signal that arrives at the receiver will be

$$\tilde{\mathbf{x}} = \mathbf{x} + \boldsymbol{\delta}, \tag{9}$$

where $\tilde{\mathbf{x}} = \mathbf{x}$ in the absence of an attack (i.e., when $\boldsymbol{\delta} = 0$). We consider a limited knowledge level threat model where the adversary knows the architecture and parameters of $f(\cdot,\theta)$ but is blind to $g(\cdot,\phi)$. This constraint mimics a real-world wireless channel where an adversary may not have complete knowledge of the underlying system under attack and thus restricts the adversary to injecting an attack in the time-domain, where traditional AMC features are constructed from.

The adversary's objective is to inject $\boldsymbol{\delta}$ to change the classification of $\mathbf{x}$ using the least amount of power possible (to evade detection caused by higher powered adversarial interference [15]), thus constraining the power of the perturbation to

$$||\boldsymbol{\delta}||_2^2 \leq P_T, \tag{10}$$

where P_T is the total power budget available to the adversary for instantiating an attack. In this work, we study two particular methods to inject adversarial interference: the fast gradient sign method (FGSM) [16], in which the adversary exhausts its total power budget on a single step attack, and the basic iterative method (BIM) [17], in which the adversary iteratively

uses a fraction of its attack budget resulting in a more powerful attack at the cost of higher computational overhead.

FGSM: In this case, the adversary adds an l_2-bounded perturbation to the transmitted signal in a single step exhausting the power budget. Formally, the n^{th} perturbed received signal is given by

$$\tilde{\mathbf{x}}_n = \mathbf{x}_n + \sqrt{P_T}\frac{\nabla_{\mathbf{x}}\mathcal{L}_n(\mathbf{x}_n, \mathbf{y}_n, \theta)}{||\nabla_{\mathbf{x}}\mathcal{L}_n(\mathbf{x}_n, \mathbf{y}_n, \theta)||_2}, \tag{11}$$

where $\mathcal{L}$ refers to the cost function of $f(\cdot, \theta)$ in (7). Adding a perturbation in the direction of the cost function's gradient behaves as performing a step of gradient ascent, thus increasing the classification error on the perturbed sample. We explore the effects of various bounds on P_T in Section III-C.

BIM The BIM is an iterative extension of the FGSM. Specifically, in each iteration, a smaller l_2-bounded perturbation, $\alpha < P_T$, is added to the transmission, and the optimal direction of attack (the direction of the gradient) is recalculated. Formally, the perturbation on iteration $k+1$ for the n^{th} sample is calculated as

$$\tilde{\mathbf{x}}_n^{(k+1)} = \mathbf{x}_n^{(k)} + \text{clip}_{P_T}\left(\sqrt{\alpha}\frac{\nabla_{\mathbf{x}}\mathcal{L}_n(\mathbf{x}_n^{(k)}, \mathbf{y}_n, \theta)}{||\nabla_{\mathbf{x}}\mathcal{L}_n(\mathbf{x}_n^{(k)}, \mathbf{y}_n, \theta)||_2}\right), \tag{12}$$

where the `clip` function is defined to ensure that the additive perturbation based on α in each iteration remains within the adversary's power budget.

III. Results and Discussion

In this section, we conduct an empirical evaluation of our method. First, we overview the dataset that we use (Sec. III-A). Next, we present the efficacy of using frequency features for classification in the absence of any adversarial interference (Sec. III-B). Finally, we demonstrate the resilience of our trained models to transferable adversarial attacks instantiated in the time domain (Sec. III-C).

A. Dataset and Evaluation Setup

We employ the GNU RadioML2016.10B dataset [18] for our analysis. Each signal in the dataset, $\mathbf{x}_n$, has an SNR of 18 dB, is normalized to unit energy, and consists of a 128-length observation window modulated according to a certain digital constellation, $\mathbf{y}_n$. We focus on the following four modulation schemes: CPFSK, GFSK, PAM4, and QPSK. Each constellation set contains 6000 examples for a total of 24000 signals. In each experiment, we employ a 70/15/15 training/validation/testing dataset split, where the training and validation data are used to estimate the parameters of $f(\cdot, \theta)$ and $g(\cdot, \phi)$, and the testing dataset is used to evaluate each trained model's susceptibility to adversarial interference and transferability to resilience. In particular, the validation set is used to tune the model parameters using unseen data during the training process whereas the testing set is used to measure the performance of the fine-tuned model. We denote the training, validation, and testing datasets, consisting of either time-domain IQ points or frequency-domain feature components, as $\mathcal{X}_{tr}^t$, $\mathcal{X}_{va}^t$, $\mathcal{X}_{te}^t$, $\mathcal{X}_{tr}^\omega$, $\mathcal{X}_{va}^\omega$, and $\mathcal{X}_{te}^\omega$, respectively.

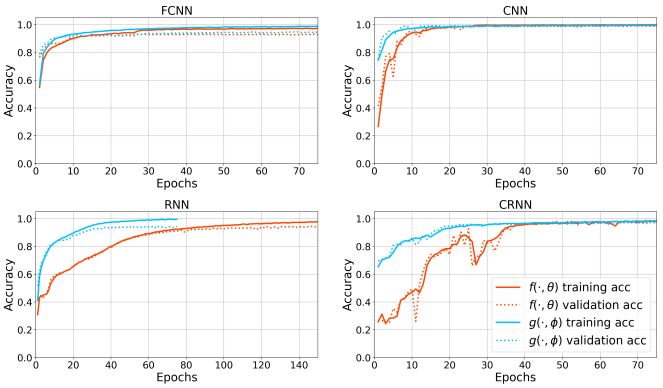

Fig. 2: The model training performance of each considered AMC architecture on the corresponding training and validation sets. We see that the frequency-based features $g(\cdot, \phi)$ outperform the time domain features $f(\cdot, \theta)$ in terms of training convergence and validation accuracy for each deep learning architecture. The CNN results in the fastest convergence and highest accuracy for both $f(\cdot, \theta)$ and $g(\cdot, \phi)$.

TABLE I: The testing accuracy of each considered model on $\mathcal{X}_{te}^{(\cdot)}$. The CNN outperforms every other considered model (although the CRNN delivers equivalent accuracy, it is achieved with a longer training time in Fig. 2 compared to the CNN).

Model	Input Features	Accuracy
FCNN	IQ	92.25%
FCNN	Frequency	92.42%
CNN	IQ	98.92%
CNN	Frequency	99.19%
RNN	IQ	93.78%
RNN	Frequency	92.67%
CRNN	IQ	98.28%
CRNN	Frequency	99.03%

B. Model Convergence Rate and Accuracy

We begin by evaluating the performance of both $f(\cdot, \theta)$ and $g(\cdot, \phi)$ in the absence of adversarial interference. In Fig. 2, we plot the evolution of the classification accuracy across training epochs achieved by each deep learning architecture on the training and validation sets. In contrast to using IQ training features, we see that each model trained using our proposed frequency feature-based input outperforms its time domain counter-part model. For example, the RNN trained on frequency components achieves an accuracy of 93.4% on its corresponding validation dataset in 75 training epochs whereas the same architecture trained on $\mathcal{X}_{tr}^t$ requires 150 epochs to converge to a validation accuracy of 93.9%. Furthermore, the CRNN also converges in fewer epochs when using frequency-based features in comparison to IQ features. We also see in Fig. 2 that the CNN obtains the best performance overall. IQ features present more challenges during training on the FCNN, RNN, and CRNN compared to the CNN. Specifically, the FCNN results in slight overfitting to the training data, the RNN fails to converge on a validation accuracy greater than 94%, and the CRNN presents instability during optimization requiring a longer number of training epochs before convergence. The CNN, on the other hand, entails almost no degree

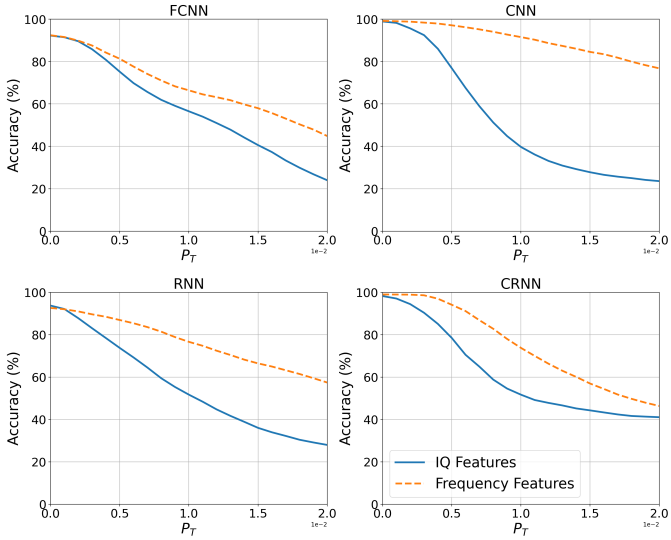

Fig. 3: The transferability of the FGSM attack from $f(\cdot, \theta)$ to $g(\cdot, \phi)$. The CNN mitigates the effects of the attack to the greatest extent with a performance improvement of 53.23% when the adversary exhausts the total perturbation budget.

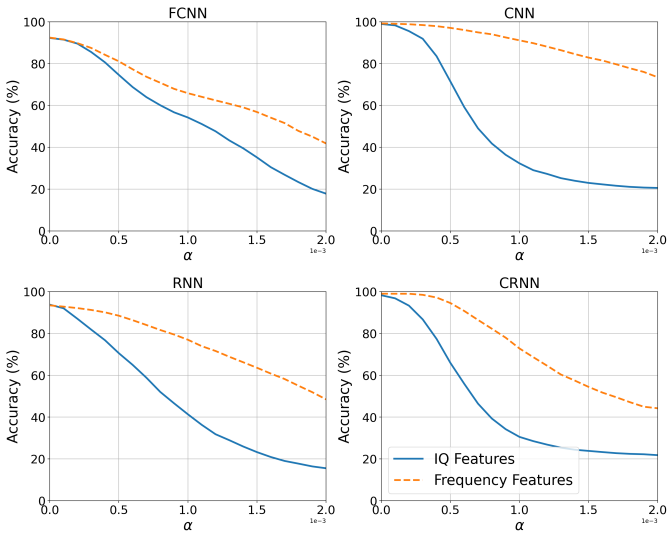

Fig. 4: The transferability of the BIM attack from $f(\cdot, \theta)$ to $g(\cdot, \phi)$. Similar to the FGSM attack, the CNN displays the strongest resilience to transferability with a performance improvement of up to 52.91%.

of overfitting while converging in substantially less epochs compared with the RNN and CRNN models.

Each trained model's accuracy on its corresponding testing set is shown in Table I. Among all eight considered models, the CNN trained on frequency features, as proposed in this work, achieves the highest testing accuracy, as well as the fastest convergence rate in Fig. 2. Specifically, this model results in nearly no overfitting between $\mathcal{X}_{tr}^{\omega}$ and $\mathcal{X}_{va}^{\omega}$, unlike either FCNN, while converging in nearly 10 epochs unlike the CNN trained on IQ features. Although the CRNN, in both cases, results in robust classification performance, the higher number of epochs required by these models results in substantially higher computational overhead (e.g., the the CNN achieves a three-fold improvement per epoch over the CRNN). *Therefore, our proposed CNN trained using frequency features is the most desirable model in terms of classification performance, training time, and computational efficiency.*

C. Model Resilience to Adversarial Interference

We now evaluate the ability of an adversarial attack instantiated in the time domain to affect our frequency domain-based AMC methodology. We begin by considering the FGSM attack where we restrict $P_T \leq 0.0200$ (corresponding to 2% additive power, which effectively degrades time domain model performance). Fig. 3 depicts the robustness of each considered model for various levels of injected interference. We see that $g(\cdot; \phi)$ improves the classification accuracy of each model in the presence of an attack on time domain feature-based classifiers. In particular, the average accuracy improvement for the FCNN, CNN, RNN, and CRNN is 10.77%, 38.32%, 20.61%, and 13.26%, respectively, across the range of P_T. The ability of the CNN and RNN to withstand attacks to the highest degree indicates their increased resilience to transferable adversarial interference.

The effect of the BIM adversarial attack is consistent with the response of the FGSM attacks. For the BIM attacks, we used ten iterations of different α-bounds with $P_T = 0.0200$. As shown in Fig. 4, the attack instantiated on time domain features is significantly mitigated on each considered model when the frequency domain is used for classification. The FCNN, CNN, RNN, and CRNN experience average improvements of 12.99%, 42.16%, 27.31%, and 27.33%, respectively, for $\alpha \in [0.000, 0.002]$. *Thus, as shown by the instantiation of both considered attacks, the transferability of adversarial interference is mitigated to the greatest extent when using the CNN as the underlying classification model.*

We analyze the performance of the CNN model more closely, both in the presence and absence of interference, in Figs. 5-7. The labels {0, 1, 2, 3} correspond to the constellations {CPFSK, GFSK, PAM4, QPSK}. As shown in Fig. 5, both time and frequency features deliver robust AMC performance in the absence of adversarial interference with classification rates of 98.92% and 99.19%, respectively. However, the classification rate in the time domain drops to a mere 23.58% and 20.56% when the FGSM and BIM perturbations are employed, respectively (where the total perturbation budget is exhausted in both cases). As shown in Figs. 6 and 7, the adversarial interference pushes the majority of signals within the classification decision boundaries of the PAM4 constellation. This is largely due to the nature of the untargeted attack in which the adversary's sole objective is to induce misclassification without targeting a specific misclassified prediction. The CNNs trained on frequency features, however, show significant improvements in classifying both FGSM and BIM perturbed signals with accuracies of 76.81% and 73.47% corresponding to classification accuracy improvements

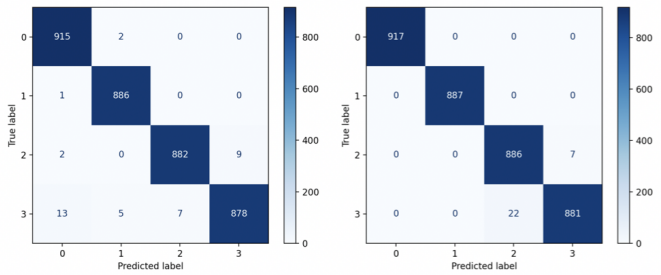

Fig. 5: The confusion matrices of the CNN's predictions with no interference, using IQ features (left) and frequency features (right). The performance for both feature representations is equivalent.

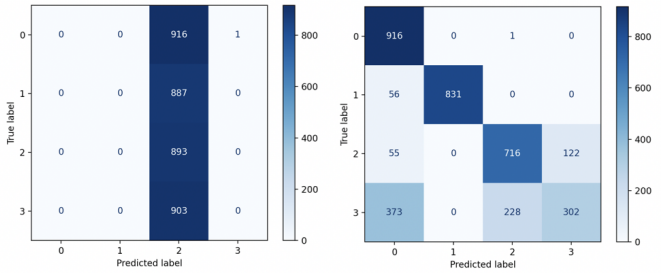

Fig. 6: The confusion matrices of the CNN classifier with the FGSM perturbation, using IQ features (left) and frequency features (right). The frequency feature-based model is able to significantly mitigate the effects of the interference induced on the IQ features.

of 53.23% and 52.91%, respectively. The frequency domain-based models correctly classify a majority of CPFSK and GFSK modulation schemes, with the largest incongruency being between PAM4 and QPSK.

IV. Conclusion and Future Work

Deep learning has recently been proposed as a robust method to perform automatic modulation classification (AMC). Yet, deep learning AMC models are vulnerable to adversarial interference, which can alter a trained model's predicted modulation constellation with very little input power. Furthermore, such attacks are transferable, which allows the interference to degrade the performance of several classifiers simultaneously. In this work, we developed a novel wireless transmission receiver architecture, consisting of a frequency domain feature-based classification model, which is capable of mitigating the transferability of adversarial interference. Specifically, we showed that our proposed frequency-feature based deep learning classifiers are resilient to transferable adversarial interference instantiated on traditional time-domain in-phase and quadrature (IQ) feature-based models. The convolutional neural network (CNN), in particular, demonstrated the most robust classification performance in the absence of an attack, along with the highest resilience to withstand additive adversarial perturbations. Future work may consider the effects of adversarial transferability in more invasive AMC environments, where the adversary's knowledge level may be unknown or unpredictable.

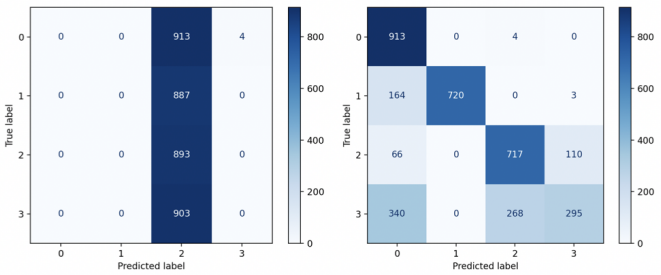

Fig. 7: The confusion matrices of the CNN's predictions with the BIM attack, using IQ features (left) and frequency features (right). Similar to the FGSM attack, the CNN trained using frequency features significantly mitigates the effects of time domain feature-based perturbations.

References

[1] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: classical approaches and new trends," *IET communications*, vol. 1, no. 2, pp. 137–156, 2007.

[2] F. Meng, P. Chen, L. Wu, and X. Wang, "Automatic modulation classification: A deep learning enabled approach," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 10 760–10 772, 2018.

[3] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv:1312.6199*, 2013.

[4] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," *arXiv preprint arXiv:1611.02770*, 2016.

[5] A. Berian, K. Staab, N. Teku, G. Ditzler, T. Bose, and R. Tandon, "Adversarial filters for secure modulation classification," *arXiv preprint arXiv:2008.06785*, 2020.

[6] 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, 2018.

[7] Y. Lin, H. Zhao, Y. Tu, S. Mao, and Z. Dou, "Threats of adversarial attacks in dnn-based modulation recognition," in *IEEE Conference on Computer Communications (INFOCOM)*, 2020, pp. 2469–2478.

[8] M. Sadeghi and E. G. Larsson, "Physical adversarial attacks against end-to-end autoencoder communication systems," *IEEE Communications Letters*, vol. 23, no. 5, pp. 847–850, 2019.

[9] Y. Arjoune and S. Faruque, "Artificial intelligence for 5g wireless systems: Opportunities, challenges, and future research direction," in *10th Annual Computing and Communication Workshop and Conference (CCWC)*, 2020, pp. 1023–1028.

[10] S. Kokalj-Filipovic, R. Miller, N. Chang, and C. L. Lau, "Mitigation of adversarial examples in rf deep classifiers utilizing autoencoder pre-training," in *International Conference on Military Communications and Information Systems (ICMCIS)*, 2019, pp. 1–6.

[11] M. Kulin, T. Kazaz, I. Moerman, and E. De Poorter, "End-to-end learning from spectrum data: A deep learning approach for wireless signal identification in spectrum monitoring applications," *IEEE Access*, vol. 6, pp. 18 484–18 501, 2018.

[12] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14 410–14 430, 2018.

[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[15] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," *arXiv preprint arXiv:1704.01155*, 2017.

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

[17] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *arXiv preprint arXiv:1607.02533*, 2016.

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