

Channel and hardware impairment data augmentation for robust modulation classification

Erma Perenda *Member, IEEE*, Gerome Bovet, Mariya Zheleva *Fellow, IEEE*, Sofie Pollin *Senior Member, IEEE*

Abstract—Deep learning has achieved remarkable results in modulation classification under two assumptions: a large amount of labeled class-balanced data is available, and the test data and training data follow the same distribution. However, due to channel and hardware impairments, it is implausible that these assumptions hold in practice. This paper proposes Model-based Data Augmentation for Deep learning-based Modulation Classification (MDA-DMC), to build a high-quality dataset from a small amount of labeled seed data. MDA-DMC leverages two well-known augmentation methods: adding Gaussian noise to, and rotation of the seed signal constellations. Furthermore, we develop two novel augmentation methods to combat channel and hardware impairments: radial shift and stretching of the signal constellations. We are the first to investigate the correlation between these augmentation methods and the channel/hardware impairments, demonstrating the adverse effect of the rotation and stretching of signal constellations on classifier performance. Consequently, the dataset must incorporate both augmentations to counterbalance performance degradation. MDA-DMC compensates for hardware impairments when training and test data channel models are identical. It also addresses fading impairments with a few AWGN seed data for low-order modulation formats. However, classifiers trained on the augmented dataset struggle to generalize channel impairments effectively with higher-order modulation formats.

Index Terms—Modulation Classification, Deep Learning, Data augmentation.

I. INTRODUCTION

AS an intermediate step between signal detection and demodulation, Automatic Modulation Classification (AMC) initially emerged within military contexts to analyze intercepted enemy signals [1]. Beyond its military applications, AMC has found extensive use in cognitive radios, where it enhances spectral efficiency and reduces receiver complexity [2], [3]. Leveraging the same advantages as in cognitive radios, AMC has been identified as the most fundamental part of intelligent transceivers for 5G and beyond networks [4] and future underwater optical wireless communications [5]. Given these communication environments' dynamic and complex nature, the importance of a reliable and impairments-resilient modulation classifier cannot be overstated.

In the literature, the AMC methods are broadly categorized into three groups: (1) Likelihood-Based (LB), Feature-Based (FB), and Deep Learning (DL)-based (DLB). LB methods treat AMC as a multi-hypothesis testing problem where the

maximum likelihood criterion is applied to the received signal directly or after some simple transformations, such as averaging [6], [7]. While LB classifiers can achieve optimal classification accuracy, they are computationally intensive and rely on the impractical assumption of perfect knowledge of signal and channel models, making them sensitive to unknown channel conditions and hardware discrepancies like Sampling Clock Offset (SCO), Carrier Frequency Offset (CFO) and In-phase/Quadrature (I/Q) imbalance. Conversely, FB methods are developed on an ad-hoc basis and lack optimality in the Bayesian sense [8], [9], [10]. These methods involve manually selecting discriminative features from raw data, such as I/Q or Power Spectral Density (PSD). This approach is labor-intensive and struggles to model all channel and hardware discrepancies, potentially leading to performance degradation [11]. Recently, DL has achieved great success in AMC due to its ability to automatically extract discriminative features using multiple hidden layers with non-linear activations [7], [12]. DLB classifiers offer higher classification accuracy and lower computational cost, making them the preferred choice among the three classifier groups.

Most of the proposed DLB classifiers [13], [14], [15], [16] achieved outstanding performance under two assumptions: (1) there is a large amount of labeled class-balanced data, and (2) the test dataset shares the same data distribution as the training dataset. Data labeling typically necessitates the presence of domain experts, leading to significant expenses. Moreover, the numerous transmitter configuration parameters and the omnipresence of various channel and hardware imperfections result in limitless data distribution variations [11]. Let us define a domain as an environment with one combination of transmitter configuration parameters, channel and hardware impairments. It is unrealistic to assume that labeled data can be acquired for each domain, as the number of domains is infinite. Many efforts have been made to enhance the robustness of modulation classifiers across various domains, encompassing advanced loss functions [17], [18], [19], more sophisticated Deep Neural Network (DNN) structures [20], [21], data augmentation techniques [22], [23] and various combinations thereof [19], [17]. Conventional data augmentation applies simple mathematical operations to signal constellations and enhances a labeled seed dataset with numerous signal distortions to model different domains. Supervised classifiers trained on the augmented dataset using a simple cross-entropy loss achieve comparable performance to other classifiers employing more advanced loss functions and complex training methods [19]. In line with the practices in image processing, AMC data augmentation has employed two well-known augmentation methods: adding Gaussian noise to, and rotation of the signal constellations [17], [22], [19]. Those operators are

E. Perenda is with DSP chair, RWTH Aachen University (email: perenda@dsp.rwth-aachen.de)

G. Bovet is with Cyber-Defence Campus, armasuisse Science&Technology (email: gerome.bovet@armasuisse.ch)

M. Zheleva is with Department of Computer Science, University at Albany - SUNY (email: mzheleva@albany.edu)

S. Pollin is with WaveCoRE, ESAT, KU Leuven (email: sofie.pollin@esat.kuleuven.be)

typically applied to randomly selected noisy and already impaired signal constellations [22], [17]. However, data augmentation performance has not received a thorough examination in the context of AMC, leaving numerous questions unanswered. This paper addresses three key questions: (1) the importance of seed data quality, (2) potential performance degradation when multiple augmentation operators are combined, and (3) the correlation between easy-to-compute augmentation operators and realistic signal impairments due to channel or hardware imperfections.

This paper proposes Model-based Data Augmentation for DLB Modulation Classification, denoted as MDA-DMC, and carries out a thorough performance evaluation to answer the above questions. MDA-DMC uses simple spatial and temporal transformations of the signal constellations to generate a domain-diverse high-quality dataset from a limited amount of labeled seed data belonging to a single domain referred to as a source domain. In addition to well-known augmentation operators (i.e., adding Gaussian noise to, or rotation of the signal constellations), we propose two novel augmentation operators named radial shift and stretching of the signal constellations. As the source domain, we choose a simple scenario consisting of an Additive White Gaussian Noise (AWGN) channel with a Signal-Noise Ratio (SNR) of 18 dB and an ideal Radio Frequency (RF) front-end. Unlike more complex channel models such as Rayleigh and Rician, the collection of labeled data for AWGN is cheaper as it does not have many hyperparameters such as path delay profiles and Doppler spread. The choice of 18 dB is driven by two considerations: (1) It mirrors a practical wireless environment (e.g., LTE) characterized by good signal quality with minor channel impairments [24], [25]; (2) Obtaining data in an environment with an SNR exceeding 18 dB, featuring excellent LTE signal quality, is likely challenging and would necessitate a testbed with an expensive isolation chamber and hardware with super low sensitivity. Notably, any SNR surpassing 13 dB (indicative of LTE good signal quality) would yield comparable performance. We selected two baseline classifiers: (1) the well-known simple 1D-Convolutional Neural Network (CNN) classifier given in [14] and (2) the more sophisticated Aggregated Residual Transformations for Deep Neural Networks (ResNeXt)-based classifier optimized by Genetic Algorithm (GA) proposed in [26]. Both classifiers utilize a straightforward cross-entropy loss, as the primary focus of this study is a comprehensive examination of data augmentation performance rather than optimizing DNN architecture. We selected simple and more sophisticated DNN classifiers to compare their ability to generalize complex and non-linear data. It is important to note that the augmented dataset can be utilized with any other DNN architecture. The key contributions of this paper are summarized below.

- This is the first detailed study of physical connection between data augmentation operators and signal impairments introduced by channel and hardware imperfections.
- We are the first to evaluate the robustness of DLB AMC to hardware impairments such as CFO, SCO and IQ-imbalance.

- We show that the proposed model-based data augmentation builds high quality dataset from a small amount of labeled seed data and significantly improves performance under different unseen channel and hardware impairments.
- We show that the quality of seed data impacts performance of data augmentation. The cleaner the seed data, the more precise the emulation of channel and hardware impairments.

The remainder of the paper is organized as follows. The related work is presented in Section II. The preliminaries, problem definition, and proposed data augmentation methods are given in Section III. The results obtained from various examined experiments are discussed in Section V. The conclusions are briefly presented in Section VI.

II. RELATED WORK

DL has achieved impressive breakthroughs in AMC [12], [7] but fails when applied to signals with unseen transmitter/channel parameters [11]. It also requires a large labeled dataset to achieve high classification accuracy. Many methods have been proposed to improve classification performance in unseen conditions, such as blind estimation of signal and channel parameters [27], [28], [29], unsupervised Domain Adaptation (DA) [30], [31], [32], [18], [33], [34], Deep Metric Learning (DML) [17], [19], Transfer Learning (TL) [11], and data augmentation. Keeping in mind the focus of this work, we summarize State-of-the-Art (SoA) achievements of data augmentation methods in what follows.

Data augmentation expands the prior knowledge by augmenting the minimally available data samples and generating more diverse samples to train the model. The simplest way to enhance the modulation dataset for different noise conditions is to add a random Gaussian noise [35], [22], [36]. Generative Adversarial Networks (GANs) have been widely used to generate additional high-quality labeled data from a small amount of seed data [37], [38], [39], [40]. One limitation of a GAN is that it cannot generate data with a distribution that differs from the existing data distribution since it attempts to learn the feature distribution of the existing data. In contrast, Spatial Transformer Network (STN) learns spatial transformations and generates additional data which might have a different distribution [11]. In [23], data is enhanced through the flip operations designed for I/Q signal data characteristics. Two flip operations are proposed: (1) a left-right flip done along the center of the time axis, and (2) an up-down flip along the origin of I/Q coordinates. In addition to adding Gaussian noise and flipping, the rotation method is introduced in [22], showing that the rotation augmentation method outperforms flipping. The authors randomly select 12.5% of data from the dataset, incorporating noisy samples with SNR ranging from -20 dB to 20 dB. In this context, adding Gaussian noise is anticipated to yield suboptimal results, given that introducing noise to instances already corrupted by noise could lead to a higher presence of instances with lower SNRs within the dataset. Nevertheless, the evaluation of augmentation methods is conducted on a dataset featuring a combination of channel

and hardware impairments. Consequently, it becomes challenging to discern the correlation between impairments and the proposed augmentation methods.

In scenarios where unlabeled data is available, one can adopt pseudo-labeling, a technique that involves assigning labels to such data based on the model's predictions, as demonstrated in [41]. Before pseudo-labeling, the feature set, consisting of 10 hand-crafted features and 30 AutoEncoder (AE)-learned features, undergoes optimization to remove redundant and irrelevant features by using a fast correlation-based filter. Moreover, [41] assumes that a few labeled samples are available for each class at each SNR. Additionally, the applied policy for pseudo-labeling cannot guarantee that the selected label is correct, especially when applied to instances with unknown channel and hardware imperfections due to their substantial distribution shifts [11].

III. METHODOLOGY

This section describes the signal model fed to the classifier's input, problem statement, preliminaries, and proposed data augmentation methods.

A. Signal model

This paper considers a Single-Input, Single-Output (SISO) system over a dynamic wireless fading channel modeled with an impulse response $h(t; \tau)$, in complex baseband equivalent notation. The $h(\cdot; \cdot)$ is a complex-valued function, τ represents the path delays of the multipath wireless channel, and t is the time variable. The input to the SISO system is a vector of complex symbols $\mathbf{a} \in \mathbb{C}^{N_s}$, where N_s denotes the number of samples per symbol. The symbols are encoded by adopting modulation format m from a pool of known modulations \mathcal{M} , shaped with a pulse of duration T_s and upconverted to center frequency f_c , forming the real transmitted passband signal $s(t)$. The output of the SISO system is the down-converted complex baseband signal, $r(t)$, which is distorted and noise-corrupted and given as

$$r(t) = s(t - \Delta t) \otimes h(t; \tau) e^{j(\phi_0 + 2\pi\Delta f t)} + v(t), \quad (1)$$

where \otimes denotes convolution in the time domain, Δt is a random time asynchronism between the transmitter and receiver clocks, Δf is the carrier frequency offset, ϕ_0 is the phase offset, and $v(t)$ is AWGN with mean 0 and variance $2\sigma_v^2$. The received signal, $r(t)$, is sampled with Nyquist frequency $1/T_r$, and N_r raw I/Q samples are fed to a modulation classifier's input. The N_r raw I/Q samples are referred to as an instance, represented as a two-dimensional array, \mathbf{r} , with dimensions $2 - by - N_r$, where the first row holds I values, and the second row holds the corresponding Q values.

B. Problem definition

This paper aims to enable robust modulation classification with limited training data for numerous combinations of channel and hardware impairments and SNR. Particularly, we target a source domain where labeled data is empirically collected for a single SNR (18 dB) and single channel (AWGN) across

all target modulations, and then augmented this baseline to match a large number of realistic cases. Let $\mathbf{r} \in \mathbb{C}^{N_r}$ be an available seed instance and $m \in \mathcal{M} = \{1, 2, \dots, M\}$ be its output label, where M is number of modulation classes. The source domain, denoted by \mathcal{D}_s , consists of n_s labeled seed instances from \mathbb{C}^{N_r} . The data available for the source domain are enhanced by applying data augmentation operators that emulate the channel and hardware impairments to obtain the enhanced dataset \mathcal{D}_s^a with $n_a > n_s$ labeled instances. The size n_a depends on how many seed instances are augmented with a range of possible augmentation methods, as will be detailed below.

Given the enhanced labeled dataset \mathcal{D}_s^a , the objective of a DLB modulation classifier is to learn a functional mapping $g: \mathbb{C}^{N_r} \rightarrow \mathcal{M}$. The functional mapping g can be decomposed into a feature encoder and a label predictor. The feature encoder, $z(r; \theta): \mathbb{C}^{N_r} \rightarrow \mathbb{R}^L$, takes an instance \mathbf{r} and generates an encoding vector $\mathbf{z}(\mathbf{r})$ of length L (θ denotes the parameters of the DNN architecture for feature encoding). The label predictor maps the encoding vectors to the label space \mathcal{M} . The functional mapping g is found by training the feature encoder and label predictor on the enhanced labeled dataset, \mathcal{D}_s^a , utilizing the cross-entropy loss.

C. Loss definition

The baseline classifiers are trained by adopting cross-entropy loss. Categorical cross-entropy [42] is a measure of the difference between two probability distributions. Softmax is utilized to convert the learned classification embeddings into the probability of belonging to each candidate modulation. When used as a loss function, the two underlying distributions are the predictions and the true classes of the samples. Categorical cross-entropy can be written as:

$$L_{ce} = -\frac{1}{N_B} \sum_{i=1}^{N_B} \sum_{j=1}^M m_{i,j} \cdot \log(\hat{m}_{i,j}), \quad (2)$$

where $m_{i,j}$ represents the ground truth, $\hat{m}_{i,j}$ is the prediction, M is the number of modulation classes, and N_B is the training batch size.

D. Data augmentation

CNN-based modulation classifiers learn spatial features of signals, i.e., signal constellations [11]. Dynamic fading channel and hardware imperfections introduce spatial transformation of the signal constellations, such as rotation, shifting, and scaling. Modeling the channel and hardware impairments has many degrees of freedom, making it tedious. Thus, simple mathematical operations are proposed to enhance a limited amount of labeled data in the source domain. We use four augmentation methods:

- 1) **Adding Gaussian Noise (AGN):** The received signal is distorted by adding Gaussian noise with zero mean value and random variance σ^2 . The noise variance is inversely proportional to the desired SNR level. The emulation of passing the received signal, \mathbf{r} , through an

AWGN channel with a certain SNR [dB] consists of the following steps:

- a) **Step 1:** measure the power of the received signal $\mathbf{r} = \begin{bmatrix} I \\ Q \end{bmatrix}_{2 \times N_r}$ as below,

$$P_r = \frac{1}{N_r} \sum_{k=1}^{N_r} |r_k|^2, \quad (3)$$

where $r_k = i_k + jq_k$ denotes k -th sample in r with in-phase value of i_k and quadrature value of q_k .

- b) **Step 2:** translate SNR [dB] to linear scale $SNR_l = 10^{SNR[dB]/10}$;
c) **Step 3:** generate Gaussian noise as

$$\mathbf{n} = \begin{bmatrix} n_I \\ n_Q \end{bmatrix} = \sqrt{\frac{P_r}{2 \cdot SNR_l}} \cdot \begin{bmatrix} randn(1, N_r) \\ randn(1, N_r) \end{bmatrix}, \quad (4)$$

where $randn(\cdot)$ generates $1 \times N_r$ array of white Gaussian noise samples with zero mean and unit variance;

- d) **Step 4:** add the generated noise to the received signal as below,

$$\mathbf{r}_n = \begin{bmatrix} \hat{I} \\ \hat{Q} \end{bmatrix} = \begin{bmatrix} I + n_I \\ Q + n_Q \end{bmatrix}. \quad (5)$$

- 2) **Rotation of the Signal Constellation (RSC):** Rotation emulates the impact of the phase offset. The phase offset might be introduced by fading channels or local oscillators. The phase offset impairs each point in the constellation, causing a rotation in the counterclockwise direction for a positive phase offset and a rotation in the clockwise direction for a negative phase offset. The augmented I/Q values by rotation with random angle θ are calculated as

$$\mathbf{r}_r = \begin{bmatrix} \hat{I} \\ \hat{Q} \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \cdot \begin{bmatrix} I \\ Q \end{bmatrix}. \quad (6)$$

- 3) **Stretching of the Signal Constellation (SSC):** Stretching emulates the impact of the amplitude imbalance, which occurs when the modulator's in-phase and quadrature components are not orthogonal. Noisy mixers used for the signal downconversion are the sources of the amplitude imbalance. A positive amplitude imbalance causes horizontal stretching of the constellation, while a negative amplitude imbalance causes vertical stretching. The amplitude imbalance is characterized by the amount of error in the amplitude, ϵ_r ($|\epsilon_r| > 1$). A positive amplitude-imbalanced impaired signal is given by

$$\mathbf{r}_i = \begin{bmatrix} \hat{I} \\ \hat{Q} \end{bmatrix} = \begin{bmatrix} \epsilon_r & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} I \\ Q \end{bmatrix}, \quad (7)$$

while a negative amplitude-imbalanced impaired signal is given by

$$\mathbf{r}_i = \begin{bmatrix} \hat{I} \\ \hat{Q} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & |\epsilon_r| \end{bmatrix} \cdot \begin{bmatrix} I \\ Q \end{bmatrix}. \quad (8)$$

Note that $|\cdot|$ is necessary in Eq. 8 for a negative amplitude imbalance value to ensure proper scaling along the quadrature axis.

Algorithm 1: Radial Shift of the Signal Constellation (RSSC)

Input: Instance r , rotation step $\Delta\theta$
Output: Radially shifted instance \hat{r}

```

1  $k \leftarrow 1$ 
2  $\hat{r} \leftarrow r$ 
3 foreach  $(i, q) \in r$  do
4    $\theta \leftarrow k \cdot \Delta\theta$ 
    $r_r = \begin{bmatrix} \hat{i} \\ \hat{q} \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \cdot \begin{bmatrix} i \\ q \end{bmatrix}. \quad (9)$ 
    $\hat{r}[k-1] \leftarrow (\hat{i}, \hat{q})$ 
5    $k \leftarrow k + 1$ 
6
7 return  $\hat{r}$ 

```

4) Radial Shift of the Signal Constellation (RSSC):

Radial shift emulates the impact of CFO and SCO caused by the local oscillators at the transmitter and receiver. CFO also occurs due to relative motion of the transmitter and/or receiver. This phenomenon is well-known as Doppler shift, and is directly proportional to the speed and direction of motion of the transmitter/receiver with respect to the direction of arrival of the received multipath wave [43]. CFO and SCO change the angles of points in the constellation linearly over time, causing points in the constellation to shift radially in the counterclockwise direction for a positive frequency offset and in the clockwise direction for a negative frequency offset. Although the points are radially shifted, their magnitude is unchanged [44]. The implementation of the radial shift augmentation method is described in Algorithm 1.

The ranges of SNR , θ , $\Delta\theta$, and ϵ_r are explored in the evaluation section. The optimal number of augmentations per method and the order of performing augmentation are also assessed in the evaluation section. Each augmented instance is normalized before it is added to the dataset \mathcal{D}_s^a . Figs. 1 and 2 show constellations of various modulated signals with realistic and emulated channel and hardware impairments, respectively. While the realistic and augmented constellations may appear similar, the performance evaluation aims to assess the precision of emulating different channel and hardware impairments.

IV. EXPERIMENTAL SETUP

This section will introduce the chosen classifier baselines and selected datasets for performance evaluation. The implementation details are also summarized.

1) *Datasets:* Two modulations sets are used: (1) *a Basic set*, containing $N_{mod} = 11$ modulation formats typically used in the literature: BPSK, QPSK, 8-PSK, 16/64-QAM, PAM4, GFSK, CPFSK, BFM, DSB-AM and SSB-AM; and (2) *an Extended set*, containing the basic ones and nine higher-order modulations: OQPSK, 32/128/256-QAM, 16/32/64/128/256-APSK ($N_{mod} = 20$). Both sets are synthetically generated in MATLAB as for the data augmentation analysis we need full control over various domains. Thus, the benchmark

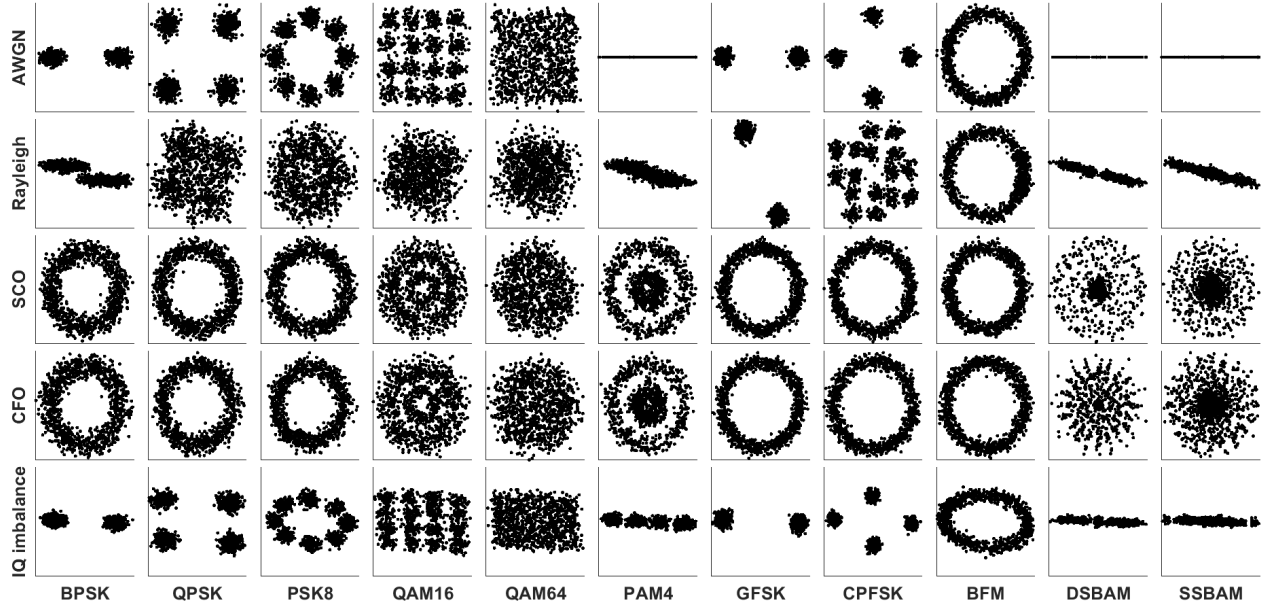


Figure 1: Constellations of several modulation types with realistic channel and hardware impairments at SNR=18 dB and upsampling factor of 1 (SC0 = 10 ppm, CFO = 15 kHz, IQ amplitude imbalance = 5 dB, IQ phase imbalance = 5 °).

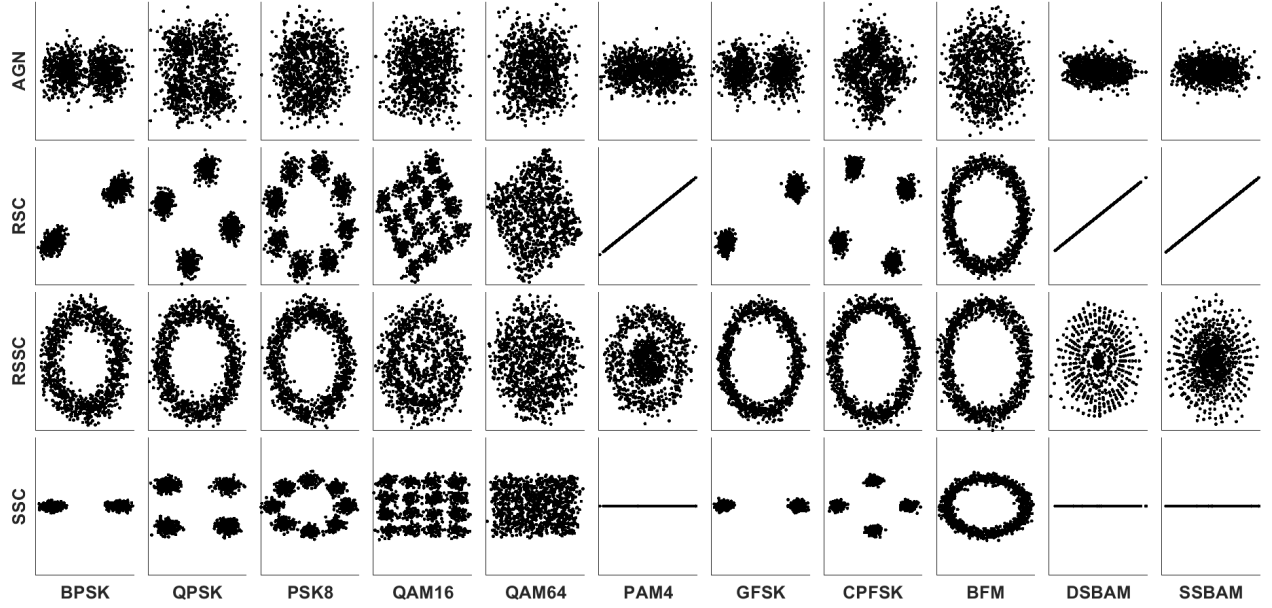


Figure 2: Constellations of several modulation types with emulated channel and hardware impairments.

datasets [45], generated for one channel model including various random hardware impairments, are unsuitable for such analysis. The code is published and available online¹. The source and target domains contain I/Q samples (instances) shaped with an upsampling factor of 4 and an Raised Cosine (RC) filter with a roll-off factor of 0.35. Instances have a size of $N_r = 128$ and $N_r = 1024$ for the basic and the extended modulation sets, respectively. The extended modulation set requires a longer signal observation because of the higher-order modulations [11]. The source domain **DS-Source** contains 100 instances for each modulation class for an 18 dB AWGN channel with ideal hardware. Each considered target domain has 1000 instances for each modulation/SNR

pair across the whole SNR range of $[-6 : 2 : 20]$ dB. The data in the source domain is available during training, while the data in the target domains is available during testing. The target domains encompass the following channel and hardware impairments:

- 1) **DS-AWGN:** AWGN with SNR ranging from -6 dB to 20 dB and the ideal RF front-end.
- 2) **DS-Rayleigh:** Rayleigh channel with a path profile: delays of $[0, 4.5, 8.5]$ μs and gains of $[0, -1, -5]$ dB. AWGN with SNR in the range of $[-6, 20]$ dB is added to the Rayleigh channel. The maximum Doppler shift is set to 4 Hz. The RF front-end is ideal.
- 3) **DS-Rician:** Rician channel with K factor of 4, a path profile with delays of $[0, 0.25, 3, 8]$ μs and gains of $[0, -2, -10, -3]$ dB. AWGN with SNR in the range of $[-6, 20]$ dB is added to the Rician channel. The

¹<https://github.com/ErmaPerenda/Modulation-dataset-generation-in-MATLAB>

maximum Doppler shift is set to 4 Hz. The RF front-end is ideal.

- 4) **DS-iqImbalance**: IQ imbalanced dataset with amplitude imbalance ranging from -10 dB to 10 dB, and phase imbalance ranging from -10° to 10° . The maximum absolute amplitude imbalance of 10 dB corresponds to a poorly designed quadrature frequency down-converter in the absence of IQ imbalance acquisition and compensation. The channel model is AWGN and the local oscillator is ideal (SCO and CFO are zero).
- 5) **DS-SCO**: Sampling clock offset dataset with a clock offset ranging from -20 ppm to 20 ppm. The maximum offset of 20 ppm corresponds to a poorly designed crystal oscillators. The channel model is AWGN, while CFO is zero and the down-converter is ideal.
- 6) **DS-CFO**: Carrier frequency offset dataset with frequency offset ranging from -10 kHz to 10 kHz. The maximum offset of 10 kHz corresponds to the performance of extremely bad CFO acquisition algorithms. The channel model is AWGN, while SCO is zero and the down-converter is ideal.
- 7) **DS-Mix**: We added random channel and hardware impairments to each instance. The fading channel is added with a probability of 70%. The fading channel type is randomly chosen, either Rayleigh or Rician, with the profiles outlined in DS-Rayleigh and DS-Rician, respectively. In contrast, SCO, CFO, and IQ imbalance impairments were addressed independently and added with a probability of 100%, allowing a single instance to undergo multiple hardware impairments. The specific values for these impairments were randomly selected from the ranges outlined in DS-SCO, DS-CFO, and DS-iqImbalance, respectively.

2) *Baselines*: This work adopts two fully-supervised classifiers to assess the data augmentation performance: (1) 1D-CNN [14] and (2) ResNeXt-based classifier optimized by GA [26]. Due to the shorter instance duration of 128 used in the basic modulation set, the two last *Convolutional* and *Pooling* layers are removed from the 1D-CNN original architecture. The original architecture of 1D-CNN is kept for the extended modulation set. The GA best-found ResNeXt architecture for the classifier consists of the feature encoder with the architecture shown in Fig. 3 and the classification head, which has one *Dense* layer with 166 dense units and *tanh* activation. The feature encoder consists of one *Convolutional* layer, four blocks with the structure shown in Fig. 4, and a *Global average pooling* layer. Each block has two parallel branches, each with two *Convolutional* layers with f filters, kernel size k , and activation a . Note that we run GA for the basic modulation set.

The alternative data augmentation techniques, GAN [40] and STN [11], claim that classifiers achieve an accuracy gain of up to 6% when trained on GAN and STN enhanced datasets. As GAN attempts to learn the data distribution of seed data, it enlarges the dataset with instances of the same distribution. GAN is mostly used to prevent classifiers' overfitting but not to combat distribution shifts due to channel and hardware

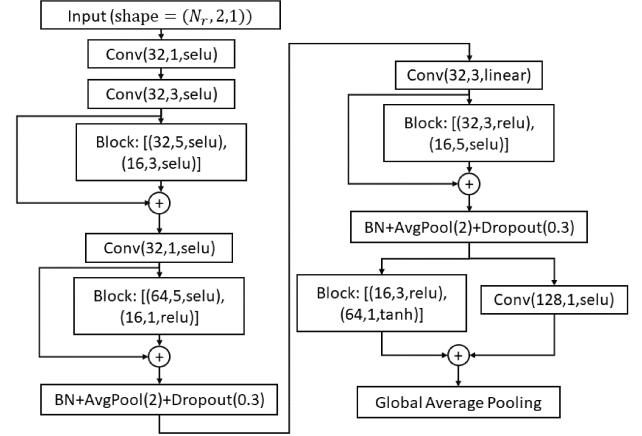


Figure 3: ResNeXt-based classifier architecture. (The Conv, BN, Dropout, and AvgPool denote *Convolutional*, *Batch Normalization*, *Dropout*, and *Average Pooling* layers, respectively. The *selu*, *relu*, and *tanh* denote Scaled exponential linear unit, Rectified linear unit, and Hyperbolic tangent activation functions, respectively.)

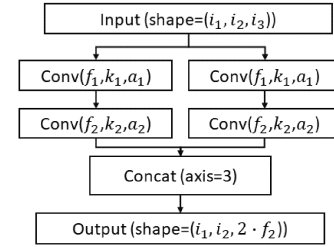


Figure 4: ResNeXt-based block structure with two parallel branches. Each branch is a serial fusion of two *Convolutional* layers.

impairments. On the other hand, in our previous work [11], we showed that STN improves accuracy by up to 6%, but it is still sensitive to distribution shifts due to channel or hardware impairments. In contrast, we proposed MDA-DMC mainly to combat distribution shifts. Therefore, a comparison of our proposed MDA-DMC with GAN- and STN-based data augmentation is out of the scope this paper.

3) *Implementation details*: All used models are implemented in TensorFlow [46]. The training is performed over $N_{epochs} = 80$ epochs and a batch size of 256 on a GPU server with eight Nvidia RTX 2080Ti cards. Adam [47] is adopted for weights' optimization with its default recommended learning rate of 0.001. Additionally, for supervised classifiers, 1D-CNN and ResNeXt, we employ the learning rate decay that reduces the learning rate by a factor of 0.2 when a validation loss has not been improved over five epochs. The code is published and available online².

V. DATA AUGMENTATION PLATEAU

In this section, we will evaluate the performance of the proposed MDA-DMC as a function of the realistic channel and hardware impairments. As already mentioned, DS-Source has 100 labeled instances for each modulation class at one

²<https://github.com/ErmaPerenda/MDA-DMC/tree/main>

SNR value (18 dB). Those 100 instances per class are referred to as the seed data in the text below. MDA-DMC consists of four augmentation operators, and we will use the basic set of modulations to explore its correlation with various channel and hardware impairments. The overall performance of MDA-DMC will be validated on the extended set of modulations.

A. MDA-DMC hyper-parameters settings

The MDA-DMC operators come with several hyper-parameters whose value ranges should be determined. We opted for the SNR range $[-6, 20]$ dB with a 2 dB increment, aligning with standard settings found in benchmark datasets. We employ a trial-and-error approach to find value ranges for other hyperparameters. The ranges are selected as a trade-off between performance gain and dataset size. We omit the detailed results for simplicity and present only the chosen ranges. RSC achieves the best results for $\theta \in [-180, 180]^\circ$ with a step of 10° . SSC is optimal for $\epsilon_r \in [-4, -1) \cup (1, 4]$ with a step 0.4. RSSC is optimal for $\Delta\theta \in [-40 : 2 : 40]^\circ$ with a step of 2° . Notably, employing smaller steps for each MDA-DMC operator has no adverse effect on classification performance; however, it significantly expands the dataset size. Given the definition of MDA-DMC operators outlined in Section III-D, one can conclude that the classification performance is not adversely affected by the order in which augmentations are applied. Next, we studied the impact of each augmentation run on dataset size and accuracy gain for each hyperparameter separately. Our experiments revealed that conducting AGN only once for each seed instance yields comparable gains compared to augmenting each seed instance multiple times for different random noise values. As the dataset size increases linearly with each run, limiting AGN augmentations to only one per seed instance makes sense. We obtained this observation after an analysis based on 100 seed instances. Applying RSC, RSSC, and SSC for each hyper-parameter value to each seed instance results in an enormous dataset, necessitating powerful GPU servers to facilitate efficient training. Initially, we began with randomly selecting one seed instance for each modulation type and augmenting it per each RSC/SSC/RSSC hyperparameter value. However, experiments demonstrated that, for a minimum of three augmentations per hyperparameter value, both classifiers effectively generalize RSC, SSC, and RSSC-augmented instances. Performing more augmentations per RSC/SSC/RSSC hyperparameter value does not yield significant accuracy improvements but drastically increases dataset size. Therefore, we chose to execute RSC/SSC/RSSC on three randomly chosen seed instances for each corresponding hyperparameter value, i.e., the total number of augmented instances is equal to $3 \cdot M \cdot (N_{RSC} + N_{SSC} + N_{RSSC})$, where M is number of modulation classes and N_{RSC} , N_{SSC} , and N_{RSSC} denote number of hyper-parameter values for RSC, SSC, and RSSC, respectively. Two augmented instances are allocated for training and one for validation.

B. Correlation between MDA-DMC operators and channel/hardware impairments

First, we evaluate the robustness of the classifiers for different channel and hardware impairments when they are

trained only on the labeled seed data from the source domain (DS-Source). We refer to this case as the starting case in the text below. Second, we compare performance gains due to the artificially adding noisy instances per each seed instance across the considered SNR range (i.e., $SNR \in [-6 : 2 : 20]$ dB). Next, we apply the other three MDA-DMC operators on the AGN-augmented dataset and evaluate classifier performance on the newly generated datasets. The results are summarized in Tables I and II. The accuracy values for target domains, DS-AWGN, DS-Rayleigh and DS-Rician are averaged over the entire SNR range, $[-6 : 2 : 20]$ dB. The accuracy values for DS-SCO, DS-CFO and DS-iqImbalance are averaged over the entire SCO, CFO and amplitude imbalance ranges, respectively, for 0 or 18 dB.

The amount of labeled seed instances (DS-Source) is not sufficient for both classifiers to generalize well, leading to notably poor performance in each channel model, as indicated in the first row in Table I. By artificially adding one noisy instance per seed instance for each SNR value in the range $[-6, 20]$ dB with a step of 2 dB, AGN augmentation built the dataset for which ResNeXt [26] increases the average accuracy values by 35.41%, 19.73%, and 19.88% in the AWGN, Rayleigh and Rician channels, respectively (the second row in Table I). On the other hand, the 1D-CNN [14] improves the average accuracy values by 16.03%, 13.99%, and 22.69% in the AWGN, Rayleigh and Rician channels, respectively. It is worth noting that RSC, RSSC, and SSC did not yield any significant accuracy gains in the AWGN channel. However, ResNeXt boosts the accuracy when trained on the RSC-augmented dataset by an additional 13.52% and 15.78% in Rayleigh and Rician channels, respectively. The RSC-augmented dataset is more complex, making it challenging for 1D-CNN to capture such complex non-linearities, thereby achieving only modest accuracy gains of up to 3.8% in fading channels. While the RSSC-augmented dataset enables the classifiers to achieve a slight accuracy improvement in fading channels, the SSC-augmented dataset confuses the classifiers more, leading to adverse effects with accuracy drops of 7% in the Rayleigh channel and 2.4% in the Rician channel.

In order to understand the impact of the hardware impairments on classifier performance, we trained ResNeXt [26] on a large amount of labeled AWGN data for the entire SNR range with the ideal RF front-end. The ResNeXt is then tested on DS-SCO, DS-CFO and DS-iqImbalance datasets. Fig. 5 shows that even a minor SCO value of ± 2 ppm results in significant drops in accuracy, specifically 61%, 56%, 40%, and 26% at SNR of 18 dB, 12 dB, 6 dB and 0 dB, respectively. The CFO value of ± 2 kHz yields nearly identical accuracy drops as the SCO value of ± 2 ppm. In contrast, ResNeXt can tolerate the amplitude imbalance of ± 5 dB. When subjected to an amplitude imbalance of ± 10 dB, ResNeXt experiences accuracy drops of 13%, 2.46%, 1.38% and 21.86% at SNR of 18 dB, 12 dB, 6 dB and 0 dB, respectively. 1D-CNN [14] follows the same accuracy drop trends as ResNeXt [26].

Table II shows that adding noisy instances improves the accuracy in the presence of hardware impairments at SNR=0 dB for both classifiers. Conversely, when we examine the SNR of 18 dB, which matches the seed SNR, we observe distinct

Table I: Correlation between MDA-DMC operators and channel impairments: accuracy gains/drops (%) versus the starting case in different channels for the basic modulation set. The first row contains the reference values for dataset size and average accuracy values (%) in the brackets for the starting case.

Trained for AWGN and				Tested for					
SNRs [dB]	Seed data per mod./SNR pair	Augmentation	Dataset size increase (#x)	DS-AWGN		DS-Rayleigh		DS-Rician	
				1D-CNN	ResNeXt	1D-CNN	ResNeXt	1D-CNN	ResNeXt
[18]	100	None	1 (1 100)	0 (46.74)	0 (46.05)	0 (39.4)	0 (32.78)	0 (28.87)	0 (25.03)
[18]	100	AGN	14	+16.03	+35.41	+13.99	+19.73	+22.69	+19.88
[18]	100	AGN+RSC	42	+19.23	+34.99	+17.54	+33.25	+26.57	+35.66
[18]	100	AGN+SSC	42	+18.57	+36.73	+12.96	+12.74	+21.19	+17.44
[18]	100	AGN+RSSC	42	+21.83	+34.59	+12.4	+28.28	+16.62	+21.45

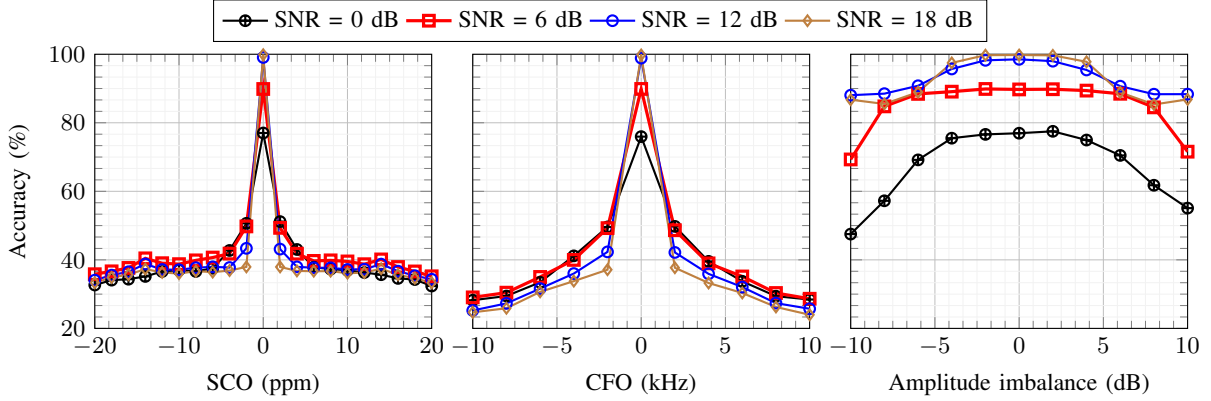


Figure 5: ResNeXt [26] sensitivity to the hardware impairments for the basic modulation set: SCO (left), CFO (middle), and IQ imbalance (right) in AWGN at different SNRs.

behaviors from ResNeXt and 1D CNN. Specifically, ResNeXt experiences a marginal accuracy drop of up to 5.63% for SCO and CFO-impaired data compared to the starting case. In contrast, 1D-CNN exhibits a slight accuracy improvement of up to 4.50%. Both classifiers have an accuracy gain of up to 31.08% for IQ imbalance-impaired data at SNR=18 dB. However, Fig. 5 shows that the classifiers are robust to IQ imbalance at high SNR values. Therefore, the AGN augmentation method cannot combat the impact of hardware impairments as it only eliminates a lack of noisy data. Table II shows that a leading contributor to combating SCO and CFO is RSSC, as it yields substantial accuracy gains of 40% and 16% for both classifiers at SNR levels of 18 dB and 0 dB, respectively. On the other hand, SSC provides accuracy improvements of 6.30% and 7.87% for IQ imbalance-impaired data at 18 dB for ResNeXt and 1D-CNN, respectively. However, it is noteworthy that RSC has a detrimental effect on IQ imbalance-impaired data, causing a significant accuracy drop of 20% at 18 dB for ResNeXt.

In conclusion, AGN serves as a countermeasure against the impact of noise, while RSC, RSSC and SSC combat fading channels, SCO/CFO, and IQ imbalance effects, respectively. Interestingly, classifiers trained on the RSC-augmented datasets experience worse performance when tested with IQ imbalance-impaired data. In contrast, classifiers trained on the SSC-augmented datasets experience accuracy drops when tested in fading channels. In what follows, we will examine how those augmentation methods work jointly and whether the adverse effects can be alleviated by achieving a balance between RSC and SSC augmented instances within the dataset.

C. Overall performance of MDA-DMC

We can treat each data augmentation type and all their possible combinations as distinct domains. We split the data augmentation process into two stages to establish balance among the domains. In the first stage, RSC, SSC, and RSSC are applied only to the AGN-augmented instances. In the second stage, each augmentation method is applied to instances that the other two methods have augmented, i.e., RSC augments the SSC- and RSSC-augmented instances; SSC augments the RSC- and RSSC-augmented instances, and RSSC augments the SSC- and RSC-augmented instances. RSC has 35 angle values, SSC has 18 ϵ_r values, and RSSC has 40 $\Delta\theta$ values. The number of augmentations for each type and modulation/SNR pair should be balanced with the number of seed instances. In our initial setup, there are 100 seed instances, with 90 allocated for training and 10 for validation. To find the optimal augmentation process, we executed four experiments wherein we applied the AGN augmentation (1) one, (2) two, (3) three, and (4) four times to each seed instance per each modulation/SNR pair. In the first stage of data augmentation, each augmentation type selects (1) one, (2) three, (3) four, and (4) six instances augmented with AGN for each modulation/SNR pair in experiments 1-4, correspondingly. In the second phase of data augmentation, each of these three augmentation methods selects (1) one, (2) one, (3) two, and (4) two instances that were previously augmented by the other two methods, corresponding to experiments 1-4, respectively. Since SSC offers only half the potential augmentations per instance compared to RSC and RSSC, we have two options to balance them. We can either decrease ϵ_r step to 0.2 or

Table II: Correlation between MDA-DMC operators and hardware impairments: accuracy gains/drops (%) versus the starting case in AWGN at SNR=0 dB and SNR=18 dB with different hardware imperfections for the basic modulation set. The first row contains the reference values for average accuracy values (%) in the brackets for the starting case.

Trained for AWGN and	Tested for											
	DS-SCO				DS-CFO				DS-iqImbalance			
	1D-CNN		ResNeXt		1D-CNN		ResNeXt		1D-CNN		ResNeXt	
	0dB	18dB	0dB	18dB	0dB	18dB	0dB	18dB	0dB	18dB	0dB	18dB
None	0 (13.01)	0 (29.11)	0 (13.38)	0 (43.32)	0 (12.57)	0 (31.97)	0 (14.92)	0 (50.61)	0 (25.54)	0 (59.43)	0 (16.14)	0 (63.32)
AGN	+15.22	+4.50	+25.31	-0.56	+17.36	+4.10	+25.76	-5.63	+24.95	+4.01	+51.01	+26.50
AGN+RSC	+22.12	+12.91	+26.04	+3.32	+25.59	+14.55	+26.31	+0.66	+25.31	-0.80	+48.75	+6.85
AGN+SSC	+16.70	+7.07	+24.81	-0.44	+19.82	+8.14	+25.28	-7.17	+29.07	+11.88	+51.28	+32.18
AGN+RSSC	+30.13	+44.68	+42.24	+39.53	+30.53	+39.80	+39.99	+32.15	+29.89	+19.85	+47.22	+23.29

Table III: Number of instances per modulation/SNR pair for each MDA-DMC augmentation method in different domain-balanced experiments.

Experiment	AGN	RSC	SSC	RSSC
#1	90	105	54	120
#1a	90	105	96	120
#1b	90	105	108	120
#2	180	175	90	200
#3	270	280	144	320
#4	360	350	180	400

double the number of augmentations for both stages of data augmentation. We explored both possibilities using the same AGN settings as in experiment 1. In particular, experiment #1a denotes the scenario where SSC adopts an ϵ_r step of 0.2. In contrast, experiment #1b maintains the same ϵ_r step of 0.4 as experiment #1, but doubles the number of SSC augmentations for both data augmentation stages. The SSC setting with the best performance is selected and applied in experiments 2-4. The number of instances per modulation/SNR pair for each augmentation method in each experiment is summarized in Table III. We compare these experiments to the baseline scenario, where 1000 labeled instances are accessible for each modulation/SNR pair in AWGN for the basic modulation set, involving both ResNeXt [26] and 1D-CNN [14]. Subsequently, we assess the most effective augmentation approach for the extended modulation set. The results are summarized in Tables IV and V.

The first three rows in Table IV show that it is better to have fewer SSC augmentations compared to RSC and RSSC augmentations. Let us compare the obtained accuracy values for experiments 1, 1a, and 1b. The classifiers achieve the highest accuracy gains in AWGN and fading channels for experiment 1 with the ϵ_r step of 0.4, where the number of SSC-augmented instances is half compared to the number of RSC- and RSSC-augmented instances. With more SSC augmentations in the training dataset, the classifiers focus more on capturing IQ imbalance while violating the discriminative features for channel impairments. As we observed in the earlier analysis, the classifiers tested on SCO and CFO-impaired data appear to be relatively insensitive to the presence of SSC augmentations in the training dataset. Both classifiers achieve slightly higher accuracy for IQ imbalance impairments for the experiments with ϵ_r step of 0.4 (see the first three rows in Table V) than the step of 0.2. As a trade-off between classifier robustness to channels and hardware impairments, for experiments 2-4, we opted for the SSC settings used in experiment 1 ($\Delta\epsilon_r=0.4$). By increasing the number of instances

per modulation/SNR pair for the augmentation methods from 1 to 6 (experiment 1 and experiment 4), the dataset size expands by a factor of 3.5. With this larger dataset, ResNeXt achieves a maximum accuracy gain of up to 4.36% for the Rician channel, whereas 1D-CNN achieves a maximum accuracy gain of 7.35% for CFO impairments. Since experiment 2 features a training dataset size that closely matches the baseline scenario's training dataset size, we will proceed to assess their performance for both the basic and extended modulation sets. The augmented dataset from experiment 2 will denote MDA-DMC augmented dataset (i.e., \mathcal{D}_s^a) in the text below.

The augmented dataset size for both modulation sets is approximately 80% of the dataset size in the baseline case. In the baseline case, training is conducted for AWGN with the entire SNR range with the ideal RF front-end and experiences high accuracy drops when tested for different channel and hardware impairments. The proposed MDA-DMC aims to achieve the same accuracy in AWGN with the ideal RF front-end as the baseline case while increasing the accuracy for different channel and hardware impairments. In all tested scenarios with the augmented dataset, ResNeXt performs better than 1D-CNN. The augmented dataset comprises highly complex data, requiring a more complex and deeper DNN architecture to capture these complexities effectively. Therefore, the subsequent analysis will focus solely on the comparison for ResNeXt.

Compared to the baseline case, ResNeXt with the augmented dataset has a lower accuracy by 6.69% and 7.88% in AWGN with the ideal RF-front end for the basic and the extended set of modulations, respectively. On the other hand, it has a higher accuracy by 13.50/16.81% and 17.20/11.98% for the basic/extended modulation set in the Rayleigh and Rician channels, respectively. Data augmentation significantly improves the accuracy by 13.38/19.02% and 42.36/38.82% for the basic/extended modulation set for SCO-impaired data at SNR of 0 and 18 dB, respectively. Similarly, the accuracy gains of 12.29/16.89% and 44.57/33.77% are achieved for the basic/extended modulation set for CFO-impaired data at SNR of 0 and 18 dB, respectively. In contrast, compared to the baseline case ResNeXt with the augmented dataset has a lower accuracy by 2.21% and 5.58% for the basic modulation set and IQ imbalance-impaired data at SNR of 0 and 18 dB, respectively. On the other hand, for the extended modulation set with IQ imbalance-impaired data, it has a higher accuracy by 3.37% at SNR of 18 dB while a lower accuracy by 5.96% at SNR of 0 dB. The proposed data augmentation methods enhance the classifiers' resilience against SCO and CFO impairments,

Table IV: MDA-DMC overall performance for different settings: accuracy gains/drops (%) versus the baseline case in different channels for the basic (BD) and extended (ED) modulation sets. The first/forth row contains the reference values for dataset size and average accuracy values (%) in the brackets for the basic/extended modulation set and the baseline case.

Trained for AWGN and					Tested for					
Exp.	SNRs [dB]	Augm.	Dataset	Total training dataset size (%)	DS-AWGN		DS-Rayleigh		DS-Rician	
					1D-CNN	ResNeXt	1D-CNN	ResNeXt	1D-CNN	ResNeXt
Baseline	[-6:2:20]	No	BD	100 (154 000)	0 (85.11)	0 (86.88)	0 (59.77)	0 (50.88)	0 (49.47)	0 (40.78)
#1	[18]	Yes	BD	46.27	-21.09	-7.11	-9.35	+10.73	-1.95	+13.21
#1a	[18]	Yes	BD	51.32	-21.39	-8.00	-9.66	+9.43	-3.81	+12.14
#1b	[18]	Yes	BD	52.99	-21.33	-8.18	-10.29	+9.67	-3.80	+13.26
#2	[18]	Yes	BD	80.63	-18.92	-6.69	-6.55	+13.50	-0.85	+17.20
#3	[18]	Yes	BD	127.03	-19.20	-7.39	-5.98	+14.17	+0.95	+18.51
#4	[18]	Yes	BD	161.52	-19.10	-5.59	-6.44	+14.51	+1.12	+17.57
Baseline	[-6:2:20]	No	ED	0 (224 000)	0 (78.68)	0 (76.85)	0 (49.24)	0 (46.34)	0 (36.41)	0 (34.86)
#2	[18]	Yes	ED	85.01	-24.02	-7.88	-3.32	+16.81	+1.55	+11.98

Table V: MDA-DMC overall performance for different settings: accuracy gains/drops (%) versus the baseline case in AWGN at SNR=0 dB and SNR=18 dB with different hardware imperfections for the basic (BD) and extended (ED) modulation sets. The first/forth row contains the reference values for dataset size and average accuracy values (%) in the brackets for the basic/extended modulation set and the baseline case.

Trained for AWGN and		Tested for											
Exp.	Ds.	DS-SCO				DS-CFO				DS-iqImbalance			
		1D-CNN		ResNeXt		1D-CNN		ResNeXt		1D-CNN		ResNeXt	
		0dB	18dB	0dB	18dB	0dB	18dB	0dB	18dB	0dB	18dB	0dB	18dB
Baseline	BD	0 (30.92)	0 (33.67)	0 (39.71)	0 (39.18)	0 (33.66)	0 (35.08)	0 (39.88)	0 (36.67)	0 (64.86)	0 (84.71)	0 (67.53)	0 (92.42)
#1	BD	+9.00	+33.56	+12.74	+41.54	+5.70	+28.92	+10.56	+43.72	-12.27	-13.45	-1.21	-2.46
#1a	BD	+9.16	+33.65	+12.59	+41.29	+5.96	+29.42	+11.41	+43.41	-12.63	-13.58	-1.74	-1.81
#1b	BD	+9.28	+34.10	+11.84	+41.35	+5.91	+29.75	+11.16	+44.06	-12.29	-14.16	-3.27	-2.31
#2	BD	+11.28	+37.77	+13.38	+42.36	+7.52	+33.93	+12.29	+44.57	-8.86	-11.60	-2.21	-5.58
#3	BD	+12.46	+39.00	+13.81	+42.79	+8.69	+35.51	+13.15	+44.98	-13.02	-12.13	-2.43	-6.59
#4	BD	+14.40	+39.86	+14.36	+43.89	+10.71	+36.27	+12.99	+45.24	-10.03	-12.07	-1.19	-4.64
Baseline	ED	0 (23.16)	0 (33.72)	0 (29.62)	0 (37.43)	0 (26.34)	0 (39.73)	0 (31.53)	0 (42.61)	0 (49.03)	0 (65.71)	0 (53.83)	0 (68.61)
#2	ED	+14.09	+25.97	+19.02	+38.82	+10.27	+18.71	+16.89	+33.77	-6.3	-6.07	-5.96	+3.37

as evident in Fig. 6 compared to Fig. 5. The minor adverse effects are present for AWGN and IQ imbalance-impaired data. In comparison to the basic modulation set, both classifiers are highly sensitive to IQ imbalance for the extended modulation set (accuracy is less than 70% at 18 dB).

To investigate the origins of these ResNeXt performance fluctuations, let us analyze the confusion matrices for the extended modulation set across various testing datasets in Figs. 7–12. Compared to the baseline case, MDA-DMC exhibits misclassifications within the QAM modulation family under conditions identical to those in which seed data are captured, as shown in Fig. 7. Notably, it is intriguing to observe that 100 seed instances are sufficient for distinguishing higher-order APSK modulations. However, higher-order QAM modulations demand more seed instances to capture all symbol transitions adequately. In contrast to the baseline case, MDA-DMC facilitates accurate classifications of analog, higher-order APSK, and low-order digital modulations in Rayleigh fading. However, intra-QAM family misclassifications persist in the presence of Rayleigh fading. Figs. 8 and 9 illustrate that MDA-DMC exhibits heightened sensitivity to Rician fading at equivalent SNR. Similarly, under hardware imperfections, MDA-DMC misclassifies QAM modulation types. Conversely, MDA-DMC accurately classifies all other modulations under substantial SCO impairment of 10 ppm and CFO impairment of 5 kHz. However, under IQ imbalance, MDA-DMC experiences misclassifications in APSK and QAM due to the similarity induced by stretching, thereby adding complexity to

the classification process. The differentiation of similar complex modulations under channel and hardware impairments may require a more sophisticated DNN architecture and a specialized loss function for the feature encoder. Nevertheless, such considerations fall beyond the scope of the current study.

D. MDA-DMC performance under joint channel and hardware impairments

The analysis above assessed the performance of MDA-DMC in two cases: (1) hardware impairments in the presence of AWGN and (2) fading impairments in the presence of AWGN and with an ideal RF front end. Additionally, each hardware imperfection was examined independently. In practical scenarios, every transmitted signal encounters a variety of channels and hardware impairments during its journey to the receiver. RF-front ends at both the transmitter and receiver sides introduce several hardware imperfections, including SCO, CFO, and IQ imbalance. Hence, in the subsequent analysis, we investigated the effectiveness of MDA-DMC emulation in situations where multiple channel and hardware impairments can coincide. First, we created a labeled DS-Mix dataset, simulating joint channel and hardware impairments. For each modulation/SNR pair, we generated 1000 instances experiencing various impairments, with SNR ranging from -6 dB to 20 dB. Second, we run three scenarios depending on the dataset for which the ResNeXt classifier was trained: (1) the MDA-DMC augmented dataset, (2) the DS-AWGN dataset, and (3) the DS-Mix dataset. Third, for each scenario we

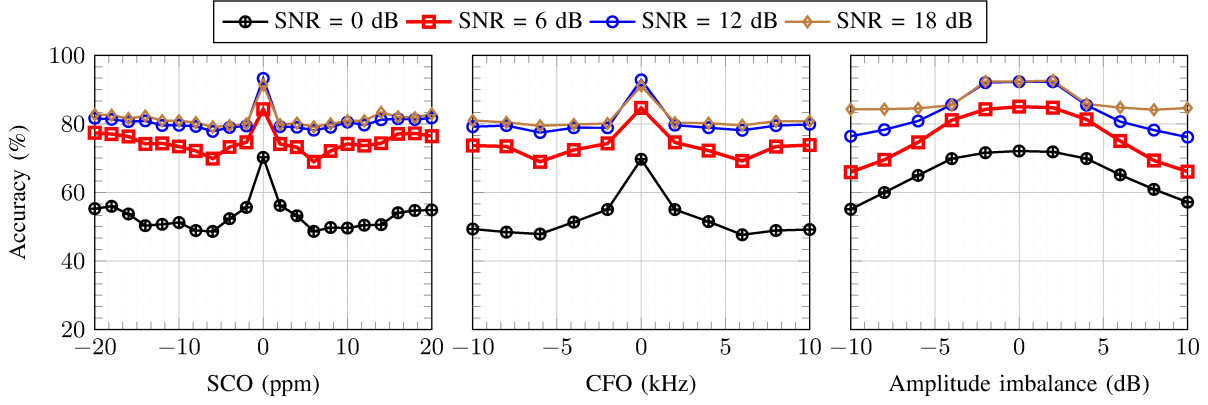


Figure 6: ResNeXt [26] sensitivity to the hardware impairments after data augmentation for the basic modulation set: SCO (left), CFO (middle), and IQ imbalance (right) in AWGN at different SNR values.

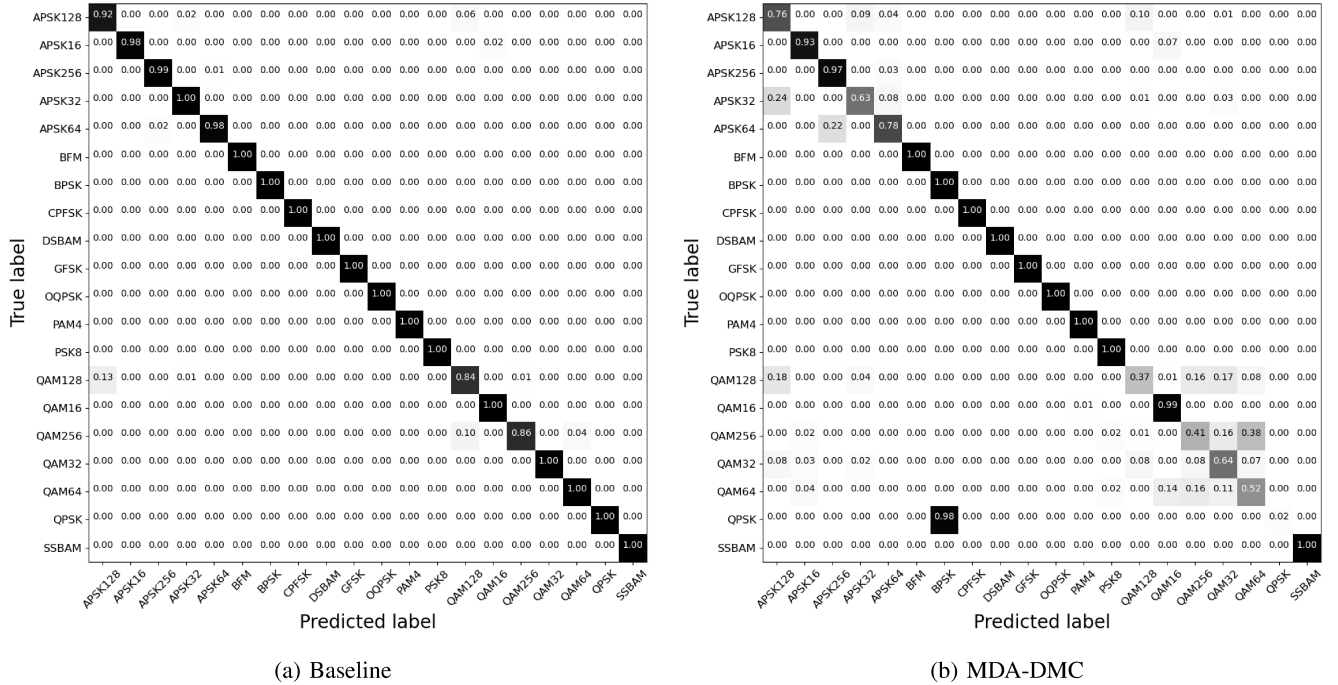


Figure 7: Confusion matrices for DS-AWGN at SNR=18 dB when ResNeXt is trained for the baseline (left) and augmented (right) datasets.

tested the trained ResNeXt classifier for the DS-Mix dataset. Each scenario was executed for both modulation sets, and the summarized results can be found in Fig. 13.

Despite DS-AWGN having 1000 labeled instances for each modulation/SNR pair, the classifier trained on it demonstrates poor generalization power, yielding an accuracy of approximately 20%. In contrast, under an ideal scenario where numerous labeled instances for joint channel and hardware impairments are assumed during the training stage (80% of the DS-Mix dataset), the classifier achieves 80% and 70% accuracy at high SNRs for the baseline and extended modulation sets, respectively. MDA-DMC generates the augmented dataset from 100 labeled instances for each modulation type at an SNR of 18 dB. The classifier trained on this augmented dataset significantly enhances accuracy compared to the baseline case (DS-AWGN). It reaches 65% and 55% accuracy at high SNRs for baseline and extended modulation sets, respectively. The observed accuracy gap of 15% compared to the classifier

trained for DS-Mix is likely attributed to MDA-DMC not emulating fading channels optimally. Consequently, further research on enhancing the emulation of fading channels is needed.

E. Does the seed data properties matter?

The seed data depends on: SNR value and number of instances per modulation. The above analysis is done for default properties' values: $SNR = 18$ dB and 100 instances per modulation. Although we justified why we chose 18 dB, we will assess how data augmentation's performance is impacted by changing the seed data properties.

1) *SNR value of seed data*: To assess the impact of SNR, we run three experiments with SNR of: (1) 0 dB, (2) 10 dB, and (3) 20 dB. We run each experiment for both basic and extended modulation sets while adopting ResNeXt as a classifier since it outperforms 1D-CNN in each evaluated scenario, as shown above. Fig. 14 shows accuracy averaged over the whole

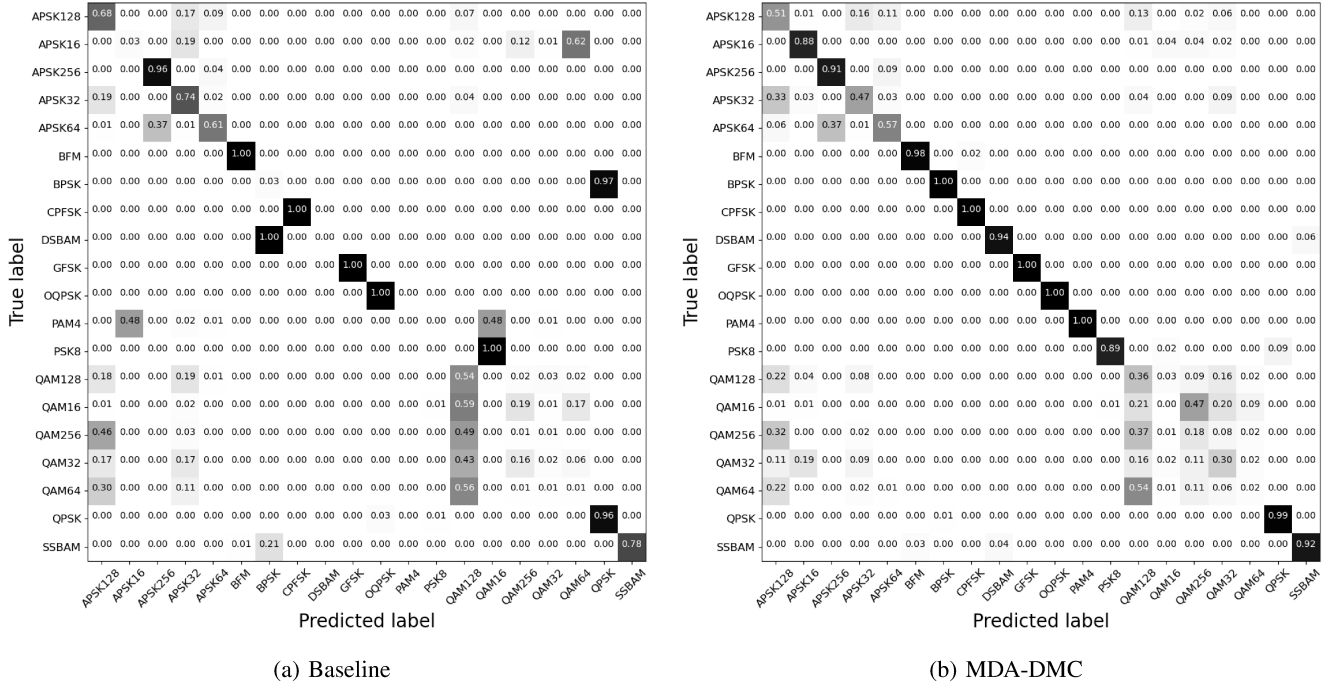


Figure 8: Confusion matrices for DS-Rayleigh at SNR=10 dB when ResNeXt is trained for the baseline (left) and augmented (right) datasets.

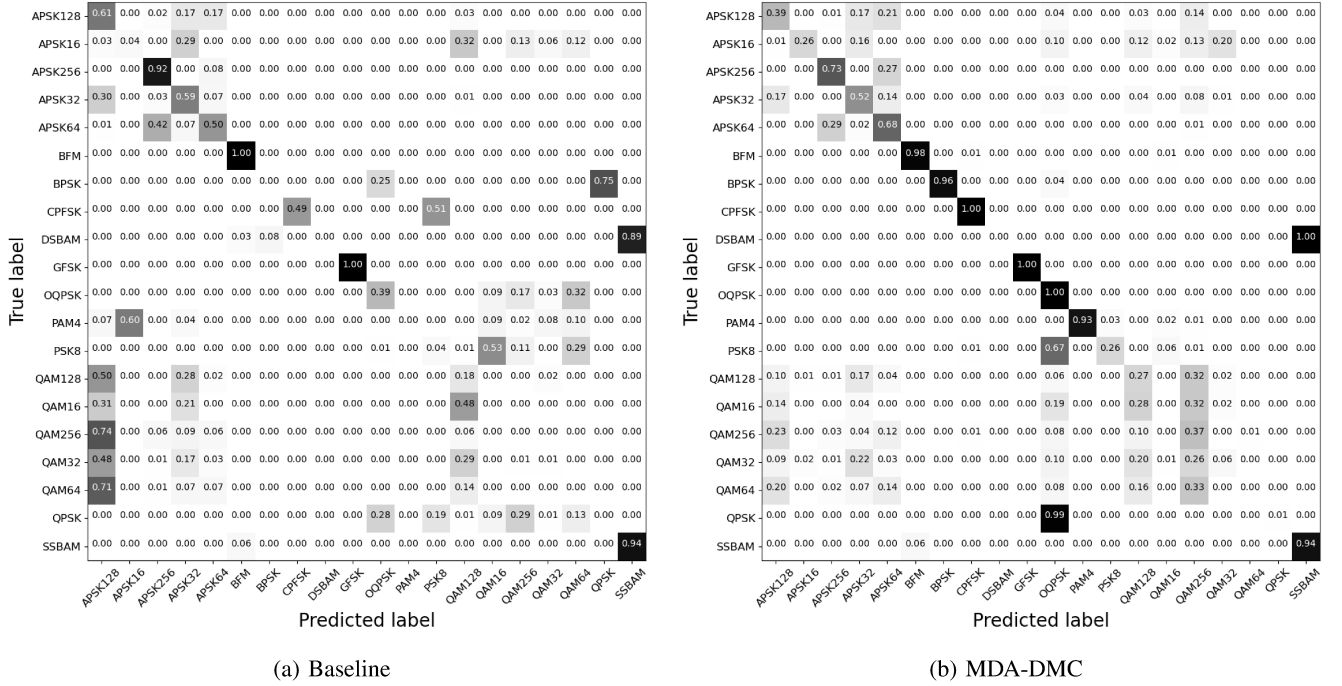


Figure 9: Confusion matrices for DS-Rician at SNR=10 dB when ResNeXt is trained for the baseline (left) and augmented (right) datasets.

SNR range for each experiment. The results show that the cleaner the seed data, the higher the accuracy for each channel and hardware impairment case, as augmentation can easily be used to add noise but not to remove it. To illustrate, consider the classifier's performance under hardware impairments at 0 and 18 dB. At 0 dB, the classifier's performance remains relatively consistent when the seed SNR exceeds 0 dB. In contrast, as the seed SNR increases to 18 dB, the classifier's

performance notably improves in the presence of hardware impairments.

2) *Seed data set size*: To assess the impact of seed data set size, we compare (1) 10, (2) 25, and (3) 50 instances per modulation/SNR pair. To keep the same dataset size, we run AGN (1) 10, (2) 4, and (3) 2 times for each seed instance for experiments 1 to 3, respectively. We keep the optimal settings for the other data augmentation methods for each experiment. We run each experiment for both basic and

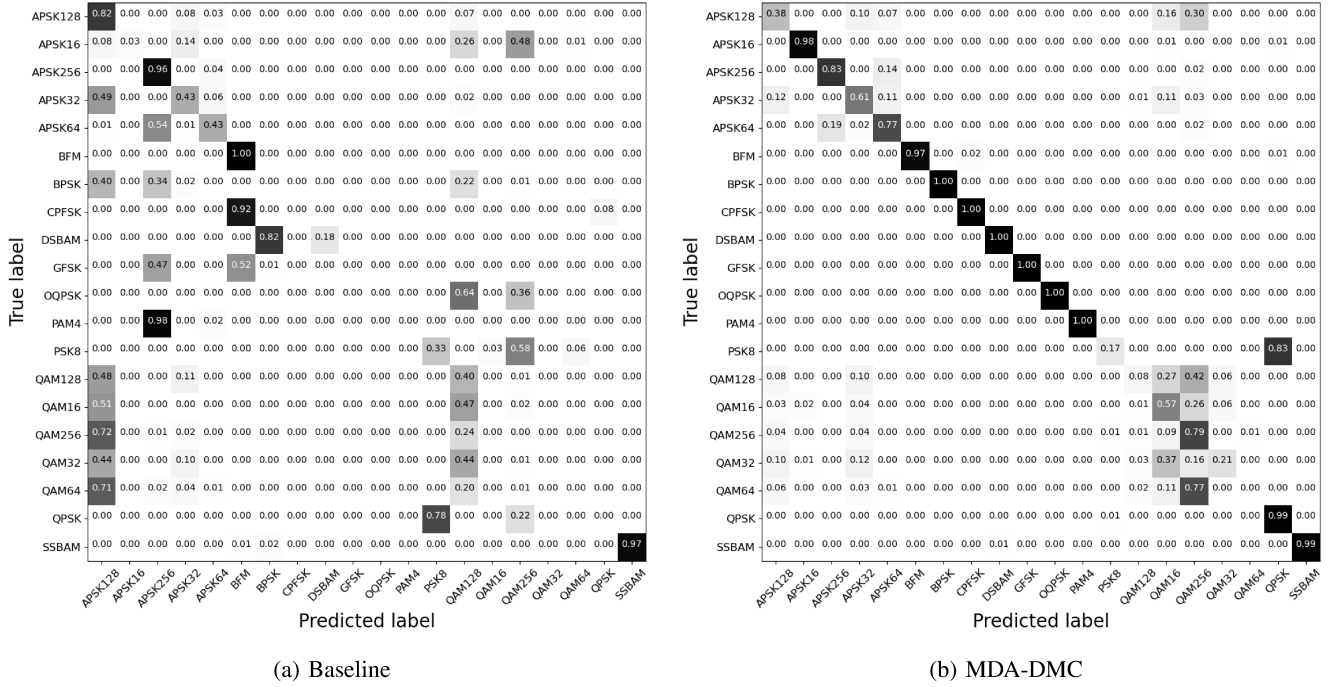


Figure 10: Confusion matrices for DS-SCO at SNR=18 dB and SCO = 10 ppm when ResNeXt is trained for the baseline (left) and augmented (right) datasets.

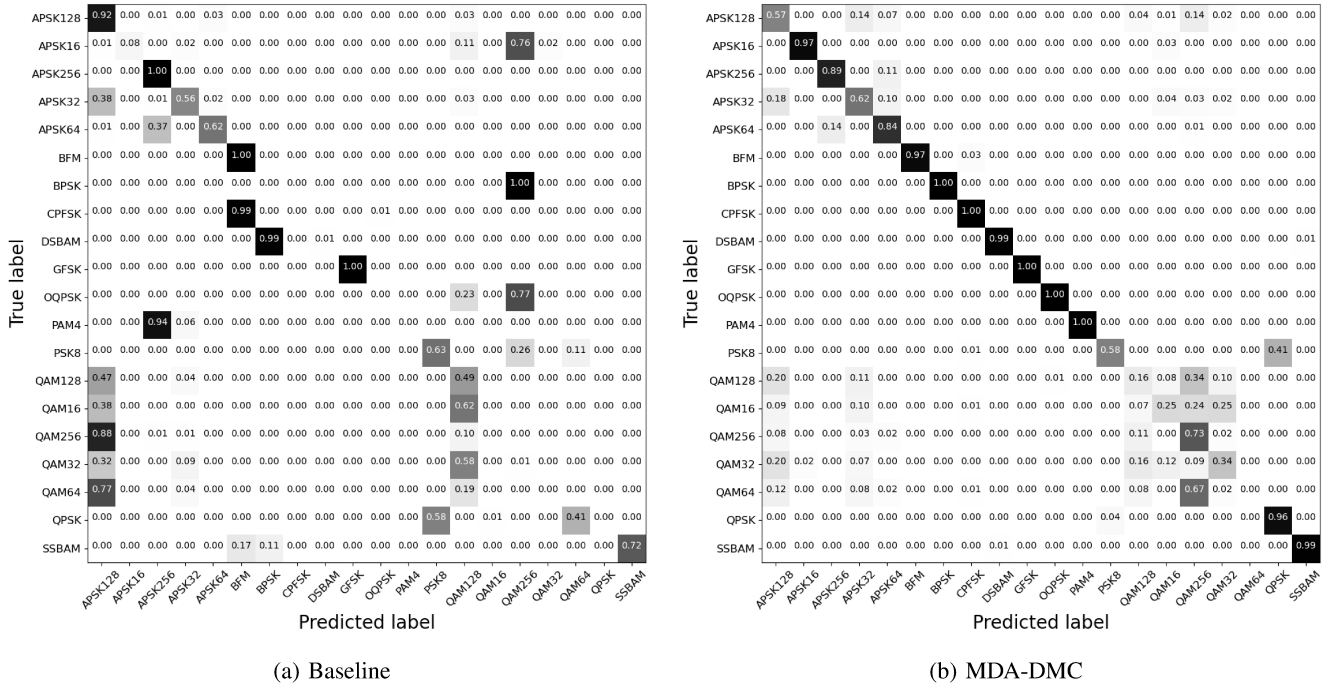
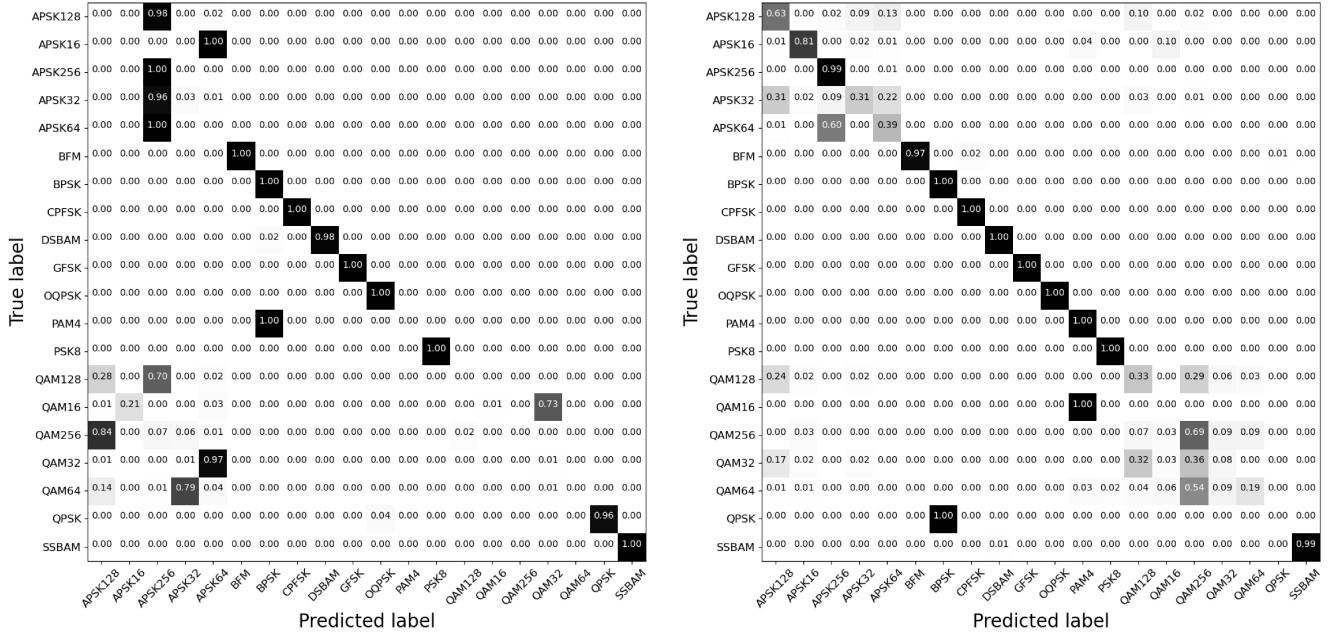


Figure 11: Confusion matrices for DS-CFO at SNR=18 dB and CFO = 5 kHz when ResNeXt is trained for the baseline (left) and augmented (right) datasets.

extended modulation sets while adopting only the ResNeXt classifier. Fig. 15 shows accuracy averaged over the whole SNR range for each experiment. MDA-DMC achieves nearly identical accuracy with 50 seed instances per modulation/SNR pair compared to 100 seed instances per modulation/SNR pair for the basic modulation set. Conversely, for the extended modulation set, accuracy experiences a slight increase as the number of seed instances increases from 50 to 100. As

demonstrated earlier, the extended modulation set demands more seed instances to distinguish higher-order QAM modulations effectively. Data augmentation performance decreases by up to 15% for channel impairments and up to 17% for hardware impairments when we decrease the number of seed instances from 100 to 10. As we showed in our previous work [11], CNN-based modulation classifiers learn both spatial features and the transitions between signal constellation points.



(a) Baseline

(b) MDA-DMC

Figure 12: Confusion matrices for DS-iqImbalance at SNR=18 dB and amplitude imbalance of 6 dB when ResNeXt is trained for the baseline (left) and augmented (right) datasets.

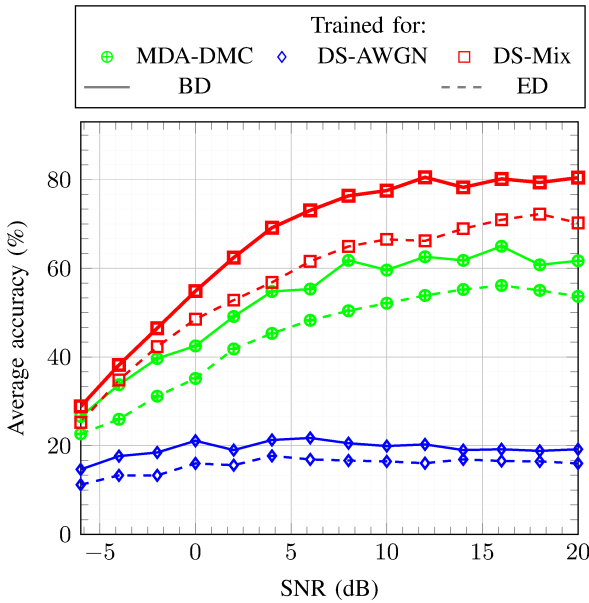


Figure 13: ResNeXt [26] performance under joint channel and hardware impairments (DS-Mix) for different training settings and both modulation sets (basic (BD) and extended (ED)).

Ten seed instances per modulation are insufficient to capture all transitions among the constellation points, especially for higher-order modulations.

VI. CONCLUSIONS

Robust modulation classification under realistic conditions is challenging primarily due to the non-linear effects introduced by channel and hardware impairments. DLB modulation classifiers demand a substantial amount of class-balanced labeled data encompassing every possible combination of channel and hardware impairments. Data augmentation emerges

as a cost-effective and practical tool to emulate diverse channel and hardware impairment scenarios. By collecting a few labeled seed data, our proposed MDA-DMC with four augmentation methods makes the classifier robust to hardware impairments with an accuracy gain of up to 40%. MDA-DMC achieves significant accuracy gains for channel impairments up to 17.20%. The results showed that MDA-DMC emulates joint channel and hardware impairments very well. Nonetheless, a 15% accuracy gap persists compared to scenarios with perfect channel and hardware impairments knowledge. While MDA-DMC brings about significant performance enhancements, future research should emphasize a deeper understanding of how fading channels impact constellation shapes. This knowledge can facilitate the development of more finely tuned augmentation methods, potentially bridging the observed performance gap.

VII. ACKNOWLEDGMENTS

This work has been partially supported by the Swiss Federal Office for Defence Procurement (armasuisse) (project code Aramis CYD-C-2020003) and NSF CNS-1845858.

REFERENCES

- [1] Z. Zhu and A. K. Nandi, *Automatic Modulation Classification: Principles, Algorithms and Applications*. Wiley Publishing, 1st ed., 2015.
- [2] Y.-C. Liang, K.-C. Chen, *et al.*, "Cognitive radio networking and communications: an overview," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 7, pp. 3386–3407, 2011.
- [3] T. Yücek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, pp. 116–130, 2009.
- [4] F. Liu, Y. Cui, *et al.*, "Integrated Sensing and Communications: Toward Dual-Functional Wireless Networks for 6G and Beyond," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 6, pp. 1728–1767, 2022.

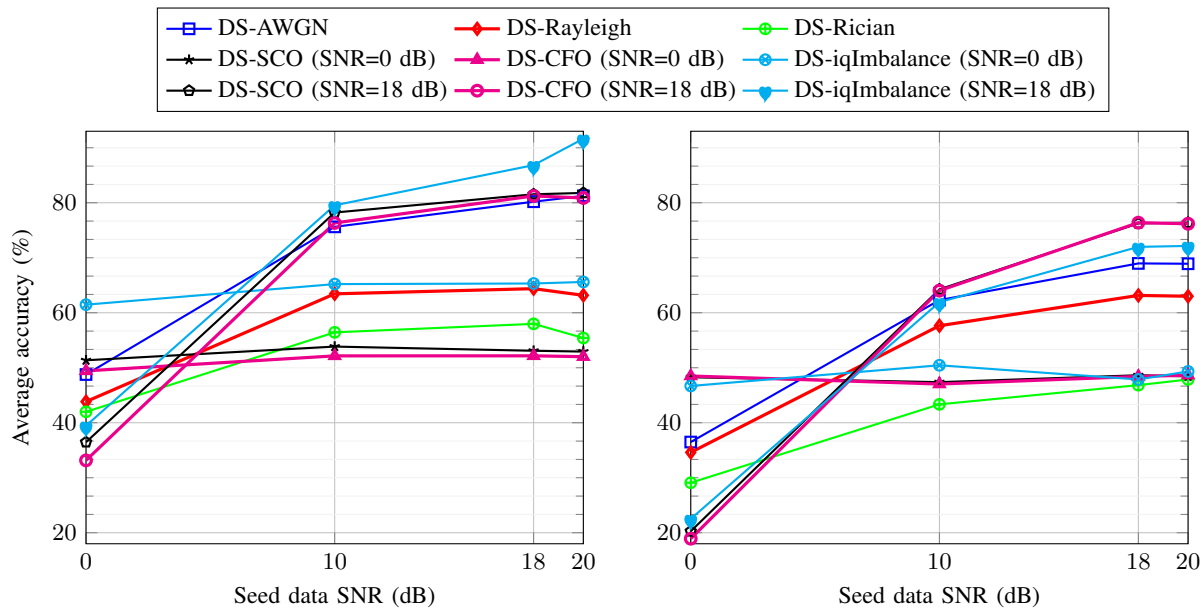


Figure 14: Data augmentation performance vs the seed data SNR value for different channel and hardware impairments evaluated for the basic modulation set (left) and the extended modulation set (right).

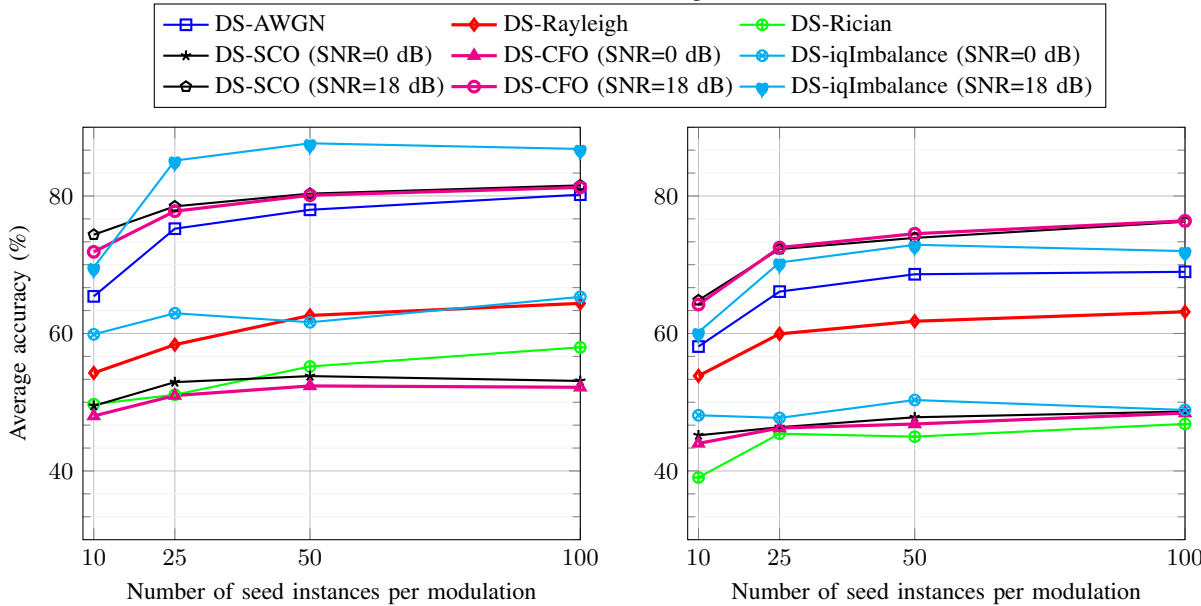


Figure 15: Data augmentation performance vs seed data set size for different channel and hardware impairments evaluated for the basic modulation set (left) and the extended modulation set (right).

- [5] L. Zhang, X. Zhou, J. Du, and P. Tian, "Fast self-learning modulation recognition method for smart underwater optical communication systems," *Opt. Express*, vol. 28, pp. 38223–38240, Dec 2020.
- [6] J. L. Xu, W. Su, and M. Zhou, "Likelihood-Ratio Approaches to Automatic Modulation Classification," *IEEE Trans. Systems, Man, and Cybernetics Part C: Applications and Reviews*, vol. 41, pp. 455–469, July 2011.
- [7] T. Huynh-The, Q.-V. Pham, *et al.*, "Automatic Modulation Classification: A Deep Architecture Survey," *IEEE Access*, vol. 9, pp. 142950–142971, 2021.
- [8] S. Majhi, R. Gupta, *et al.*, "Hierarchical Hypothesis and Feature-Based Blind Modulation Classification for Linearly Modulated Signal," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 12, 2017.
- [9] Y. Zhang, J. Wang, *et al.*, "Wireless Signal Classification Based on High-Order Cumulants and Machine Learning," in *2018 IEEE International Conference of Safety Produce Informatization (IICSPI)*, pp. 246–250, 2018.
- [10] F. Yang, B. Hao, *et al.*, "A Method of High-Precision Signal Recognition Based on Higher-Order Cumulants and SVM," in *2018 5th International Conference on Systems and Informatics (ICSAI)*, pp. 455–459, 2018.
- [11] E. Perenda, S. Rajendran, *et al.*, "Learning the unknown: Improving modulation classification performance in unseen scenarios," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, pp. 1–10, 2021.
- [12] B. Jdid, K. Hassan, *et al.*, "Machine Learning Based Automatic Modulation Recognition for Wireless Communications: A Comprehensive Survey," *IEEE Access*, vol. 9, pp. 57851–57873, 2021.
- [13] S. Rajendran, W. Meert, *et al.*, "Deep Learning Models for Wireless Signal Classification with Distributed Low-Cost Spectrum Sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, 2018.
- [14] T. 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, 2018.
- [15] Y. Ren, W. Jiang, and Y. Liu, "Complex-valued Parallel Convolutional Recurrent Neural Networks for Automatic Modulation Classification," in *2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pp. 804–809, 2022.

- [16] S. Chang, S. Huang, *et al.*, "Multitask-Learning-Based Deep Neural Network for Automatic Modulation Classification," *IEEE Internet of Things Journal*, vol. 9, no. 3, pp. 2192–2206, 2022.
- [17] D. Liu, P. Wang, *et al.*, "Self-Contrastive Learning based Semi-Supervised Radio Modulation Classification," in *MILCOM 2021 - 2021 IEEE Military Communications Conference (MILCOM)*, pp. 777–782, 2021.
- [18] Z. Ke and H. Vikalo, "Real-Time Radio Modulation Classification With An LSTM Auto-Encoder," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4935–4939, 2021.
- [19] E. Perenda, S. Rajendran, G. Bovet, M. Zheleva, and S. Pollin, "Contrastive learning with self-reconstruction for channel-resilient modulation classification," in *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications*, pp. 1–10, 2023.
- [20] L. Zhang and X. Gao, "Transfer Adaptation Learning: A Decade Survey," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–22, 2022.
- [21] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.
- [22] L. Huang, W. Pan, Y. Zhang, L. Qian, N. Gao, and Y. Wu, "Data Augmentation for Deep Learning-Based Radio Modulation Classification," *IEEE Access*, vol. 8, pp. 1498–1506, 2020.
- [23] Y. Wang, J. Bai, *et al.*, "MsmcNet: A Modular Few-Shot Learning Framework for Signal Modulation Classification," *IEEE Transactions on Signal Processing*, vol. 70, pp. 3789–3801, 2022.
- [24] Keenetic, "LTE signal quality parameters," 2022. [Online]: Available: <https://help.keenetic.com/hc/en-us/articles/360000403359-LTE-signal-quality-parameters>.
- [25] "LTE reference sensitivity — snr and throughput at the cell edge," 2022. [Online]: Available: <https://www.techplayon.com/lte-reference-sensitivity-calculation-how-it-depends-on-snr-why-throughput-decrease-at-the-cell-edge/>.
- [26] E. Perenda, S. Rajendran, *et al.*, "Evolutionary Optimization of Residual Neural Network Architectures for Modulation Classification," *IEEE Transactions on Cognitive Communications and Networking*, vol. 8, no. 2, pp. 542–556, 2022.
- [27] E. Rebeiz, F.-L. Yuan, *et al.*, "Energy-Efficient Processor for Blind Signal Classification in Cognitive Radio Networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 2, pp. 587–599, 2014.
- [28] E. Kazikli, B. Dulek, and S. Gezici, "Optimal Joint Modulation Classification and Symbol Decoding," *IEEE Transactions on Wireless Communications*, vol. 18, no. 5, pp. 2623–2638, 2019.
- [29] S. Hanna, C. Dick, and D. Cabric, "Signal Processing-Based Deep Learning for Blind Symbol Decoding and Modulation Classification," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 82–96, 2022.
- [30] K. Bu, Y. He, *et al.*, "Adversarial Transfer Learning for Deep Learning Based Automatic Modulation Classification," *IEEE Signal Processing Letters*, vol. 27, pp. 880–884, 2020.
- [31] L. Li, Q. Peng, *et al.*, "Deep Modulation Recognition in an Unknown Environment," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pp. 1045–1048, 2019.
- [32] Q. Wang, P. Du, *et al.*, "Adversarial unsupervised domain adaptation for cross scenario waveform recognition," *Signal Processing*, vol. 171, p. 107526, 2020.
- [33] J. Qiao, W. Chen, *et al.*, "Blind Modulation Classification Under Uncertain Noise Conditions: A Multitask Learning Approach," *IEEE Communications Letters*, vol. 26, no. 5, pp. 1027–1031, 2022.
- [34] Y. Wang, G. Gui, *et al.*, "Transfer Learning for Semi-Supervised Automatic Modulation Classification in ZF-MIMO Systems," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 10, no. 2, pp. 231–239, 2020.
- [35] J. A. Latshaw, D. C. Popescu, J. A. Snoap, and C. M. Spooner, "Using Capsule Networks to Classify Digitally Modulated Signals with Raw I/Q Data," in *2022 14th International Conference on Communications (COMM)*, pp. 1–6, 2022.
- [36] X. Yun and X. Zhou, "Exploring Self-supervised Learning for Radio Signal Recognition," in *2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*, pp. 2425–2430, 2021.
- [37] I. Lee and W. Lee, "UniQGAN: Unified Generative Adversarial Networks for Augmented Modulation Classification," *IEEE Communications Letters*, vol. 26, no. 2, pp. 355–358, 2022.
- [38] B. Tang, Y. Tu, *et al.*, "Digital Signal Modulation Classification With Data Augmentation Using Generative Adversarial Nets in Cognitive Radio Networks," *IEEE Access*, vol. 6, pp. 15713–15722, 2018.
- [39] S. Rajendran and S. Pollin, "Large scale wireless spectrum monitoring: Challenges and solutions based on machine learning," in *Spectrum Sharing: The Next Frontier in Wireless Networks* (D. T. M. S. Tharmalingam Ratnarajah, Constantinos B. Papadias, ed.), ch. 16, Wiley-Blackwell, 2020.
- [40] M. Patel, X. Wang, and S. Mao, "Data augmentation with conditional GAN for automatic modulation classification," *Proceedings of the 2nd ACM Workshop on Wireless Security and Machine Learning*, 2020.
- [41] Y. Shi, H. Xu, L. Jiang, and Y. Liu, "Few-Shot Modulation Classification Method Based on Feature Dimension Reduction and Pseudo-Label Training," *IEEE Access*, vol. 8, pp. 140411–140425, 2020.
- [42] S. Huang and T. Le, *Principles and Labs for Deep Learning*. Elsevier Science, 2021.
- [43] T. Rappaport, *Wireless Communications: Principles and Practice*. USA: Prentice Hall PTR, 2nd ed., 2001.
- [44] F. Horlin, S. De Rore, *et al.*, "Impact of frequency offsets and IQ imbalance on MC-CDMA reception based on channel tracking," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 6, pp. 1179–1188, 2006.
- [45] T. J. O'Shea and N. West, "Radio machine learning dataset generation with gnu radio," *Proceedings of the GNU Radio Conference*, vol. 1, no. 1, 2016.
- [46] M. Abadi, A. Agarwal, *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. Software available from [tensorflow.org](https://www.tensorflow.org).
- [47] D. P. Kingma and J. L. Ba, "Adam: A Method for Stochastic Optimization," *San Diego: The International Conference on Learning Representations (ICLR)*, vol. 1, no. 1, 2015.