IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 22 May 2023, accepted 10 June 2023, date of publication 16 June 2023, date of current version 21 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3286939

RESEARCH ARTICLE

Automatic Modulation Classification for Adaptive OFDM Systems Using Convolutional Neural Networks With Residual Learning

ANAND KUMAR[1], (Graduate Student Member, IEEE), KEERTHI KUMAR SRINIVAS[2], AND SUDHAN MAJHI[3], (Senior Member, IEEE)

[1]Department of Electrical Engineering (EE), Indian Institute of Technology (IIT) Patna, Patna 801106, India
[2]Altiostar, A Rakuten Symphony Company, Bengaluru 560102, India
[3]Department of Electrical Communication Engineering, Indian Institute of Science (IISc), Bengaluru 560012, India

Corresponding author: Sudhan Majhi (smajhi@iisc.ac.in)

This work was supported in part by the Science and Engineering Research Board (SERB), Department of Science and Technology (DST), Government of India, under Project TTR/2021/000128 and Project STR/2022/000045; and in part by the Ministry of Electronics and Information Technology, Government of India, under Project 13 (2)/2020-CC&BT.

ABSTRACT Automatic modulation classification (AMC) is becoming a promising technique for future adaptive wireless transceiver systems. The existing blind modulation classification (BMC) methods for orthogonal frequency division multiplexing (OFDM) fail to achieve the required performance by using statistical-based methods. Thus, the modulation classification research community is trying to adopt the deep learning (DL) method to improve the modulation classification accuracy. However, most of the existing DL methods for AMC of OFDM that involve the extraction of statistical features from the signal do not work for adaptive transceiver systems where the signal parameters are changed dynamically. In this paper, we design and implement AMC for adaptive OFDM systems by using a convolutional neural network (CNN) with residual learning. The proposed AMC can identify the modulation format of the received OFDM signal with different number of subcarriers, randomized carrier frequency offset (CFO), symbol timing offset (STO), phase offset, and unknown channel state information. We use residual learning to mitigate the effect of varying CFO, STO, and AWGN noise on the received OFDM signal. A larger pool of modulation schemes such as binary phase-shift keying (BPSK), quadrature PSK (QPSK), offset QPSK, $\pi/4$-QPSK, minimum shift keying, 8-PSK, 16-quadrature amplitude modulation (QAM), and 64-QAM are being considered for the proposed AMC for OFDM system in a dynamic environment. The performance and complexity of the proposed AMC are compared with the existing statistical feature-based and DL-based approaches. The proposed AMC for the OFDM system is also verified on the real-time data set generated from the universal software radio peripheral testbed setup.

INDEX TERMS Automatic modulation classification, CNN, deep learning, OFDM, residual learning.

I. INTRODUCTION

Future generation of communication systems are expected to operate in a variety of environments serving varying requirements in terms of data rate, coverage, number of connected devices, etc. This requires the next-generation communication systems to have added intelligence to interact with their environment and adapt their parameters for delivering optimal performance even in the physical layer [1], [2]. However, synchronization of these parameters across the transceiver is a highly challenging task for an adaptive or automated communication system which will be a key player for beyond fifth-generation (5G) wireless communication [3]. Explicit knowledge of these parameters between a transmitter and a receiver often results in inefficient use of available resources. For example, a transmitter that adapts its modulation scheme

The associate editor coordinating the review of this manuscript and approving it for publication was Yafei Hou.

and data rate needs to convey this information to the receiver. However, explicitly transmitting the information by using training sequence or pilot symbols reduces the spectral and power efficiency of the transmission and also increases the latency due to added computational overhead. Moreover, in severe channel conditions, training sequences may lose their property and may not be able to retrieve the parameter accurately. Hence, for the optimal use of resources and to improve spectrum efficiency, power efficiency, and latency, the receiver is expected to have some kind of intelligence built into it to detect these changing parameters. Automatic modulation classification (AMC) plays a prominent role in the development of such adaptive or intelligent transceiver systems [4]. It enables the receiver to identify the modulation format of the received signal without the aid of any explicit information of the transmitted signal parameters such as symbol rate, carrier frequency offset (CFO), symbol timing offset (STO), phase offset, and channel state information (CSI). The AMC has various applications in civilian and military communication systems and is becoming a promising automated physical layer solution in sixth-generation (6G) communication [5]. Thus, AMC has drawn the attention of many researchers in recent years. However, there is very less research work available for AMC of orthogonal frequency division (OFDM) system, which can provide better performance in a fully dynamic environment.

Many statistical methods have been proposed for the AMC of single-carrier, OFDM, and multiple input multiple output (MIMO) systems. They can be broadly classified into two main categories, i.e., likelihood-based (LB) and feature-based (FB) [4]. The LB classifiers are based on maximizing the likelihood function of the received signal with respect to the unknown modulation classes. Though LB classifiers are optimal in the Bayesian sense, their high computational complexity and analytical difficulties in modeling the unknown parameters result in sub-optimal solution [6] that affects the overall system performance. The FB classifiers use the features extracted from the received signal to identify its modulation format. Some of these features include instantaneous values of the signal [7], cumulants [8], [9], [10], [11], cyclic statistics [12], [13], [14], and wavelet transforms [15], [16], [17]. However, there are several problems with these FB approaches. The features extracted involve complex signal processing techniques and are tailored for a particular class of modulation formats. Hence, they fail to generalize well for a wide variety of modulation formats for the OFDM system. These features are mainly designed for the classifier based on some assumptions of the system parameters, such as the distribution of added noise, the number of subcarriers, null subcarrier, symbol duration, cyclic prefix (CP) length, etc. However, the modulation classification fails under the conditions when these assumptions are not satisfied. Also, when the transmitted signal gets affected by channel impairments, such as frequency offset or timing offset, the existing modulation classifier for the OFDM system cannot cope up with such a randomized environment. The cognitive radio is such adaptive technology where the parameters of the system get changed randomly from time to time. The presence of CFO in the OFDM signal destroys the orthogonality between the subcarriers leading to the loss of information contained in the signal, thereby affecting the modulation classification performance. Similarly, the presence of STO decreases the modulation classification rate in the OFDM system. Hence, the designed features for the modulation classification of the OFDM system must be resilient to these impairments and changes, and it is often a highly challenging task to design such features for OFDM systems.

Recent advancements in the field of machine learning (ML) and data science have led to its widespread application in several areas. Advanced ML techniques such as deep learning (DL) have shown a great improvement in state-of-the-art results in the fields of computer vision, speech recognition, drug discovery, and genomics [18]. The success of DL in these fields has drawn the attention of researchers to apply DL techniques to the physical layer of communications [19], [20], [21]. The work by O'Shea et al. discusses various applications of DL for the physical layer communications (PLC) [22]. Several methods have been proposed to solve the problem of AMC for single carrier systems using DL [23], [24], [25], [26], [27], [28], [29], [30], [31]. O'Shea et al. proposed the use of convolutional neural networks (CNN) for classifying various analog and digital modulation schemes for the single carrier system from the in-phase and quadrature (IQ) samples [23]. The idea was further extended to a larger class of modulation formats in [24] and was shown that the model can generalize well for a large pool of modulation classes. However, only single carrier system was considered in this work. The idea of using IQ samples and constellation diagrams as inputs to deep neural network (DNN) for the task of modulation classification for the single carrier system was discussed in [25]. However, in this DNN model, the construction of constellation diagram requires knowledge of the parameters of the transmitted signal, and thus, the method cannot be used for the AMC when such information is not available. In [26], the authors propose the use of spectral correlation function (SCF) and deep belief network (DBN) model for the AMC of single carrier system. Generation of the SCF requires pre-processing of the received signal, which increases the overall computational complexity of the AMC. The use of long short-term memory (LSTM) models for the modulation classification of single carrier systems for distributed wireless sensor networks was presented in [27]. In [28], the authors convert the raw signals into images with grid-like topology and use CNNs to identify the modulation format for single carrier system.

There are some works available for AMC of OFDM system. The work presented in [32] and [33] discusses DL-based modulation classification for OFDM systems with the knowledge of the number of subcarriers and for particular

channel environments. If any parameters such as number of subcarriers, number of null subcarriers, STO, CFO, phase offset, and CP length change, these AMC do not provide accuracy of more than 50% for adaptive OFDM systems. Zhang et al. [34] present CNN-based AMC for OFDM system. It considers only PSK modulation classes and does not present the effect of varying signal parameters on the classification performance. To the best of the authors' knowledge, there is no DL-based AMC work available for adaptive OFDM systems with a larger pool of modulations, i.e., binary phase-shift keying (BPSK), quadrature PSK (QPSK), offset QPSK (OQPSK), $\pi/4$-QPSK, minimum shift keying (MSK), 8-PSK, 16-quadrature amplitude modulation (QAM), and 64-QAM, with the varying number of subcarriers in the presence of randomized CFO, STO, phase offset, and without the requirement of channel statistics.

In this work, we propose an AMC for an adaptive OFDM system by using CNN with residual learning, which is capable of classifying the modulation format from the baseband IQ samples. Residual learning helps to overcome the degradation problem encountered in training deep networks [35]. Residual learning is also shown to be effective for the task of blind denoising of the images corrupted with unknown levels of Gaussian noise [36]. Thus, residual connections help to reduce the effect of added noise on the received OFDM signal. The main contributions of this paper are as follows

- We propose the AMC for adaptive OFDM signals for a large pool of modulation classes, i.e., BPSK, QPSK, OQPSK, $\pi/4$-QPSK, MSK, 8-PSK, 16-QAM, and 64-QAM.
- The another major contribution is that the proposed AMC can work in the presence of any number of subcarriers, randomized STO, CFO, and phase offsets under unknown frequency-selective fading channel conditions.
- The performance of the proposed AMC is compared with the existing methods and validated over a real-time dataset generated in an indoor environment by using National Instruments (NI) universal software radio peripheral (USRP). The complexity of the proposed AMC is derived and compared with the existing model.

The rest of the paper is organized as follows. The system model, along with the OFDM model and the mathematical framework of DL, is described in Section II. In Section III, we provide the details of the architecture of the neural network considered in this work. The details of the data generation process, simulation, experimental results, and computational complexity are presented in Section IV. Finally, we draw conclusions from our various experiments in Section V.

Notations

Throughout this article, lowercase bold letters denote vectors, and uppercase bold letters denote matrices. $(.)^T$ denotes the transpose of a vector or matrix. $(.)^{(i)}$ represents the ith example or instance.

II. SYSTEM MODEL

The system model of AMC for adaptive OFDM system is shown in Fig. 1. It consists of an adaptive OFDM transmitter, receiver, and CNN-based AMC. The transmitter is capable of adjusting its baseband modulation format and number of subcarriers as per the requirement of data rate and available CSI. The transmitted signal then passes through the frequency-selective fading channel. This channel subjects the transmitted signal to all kinds of impairments, such as timing, frequency, and phase offsets. The receiver consists of an AMC system comprised of a CNN. The CNN is pre-trained with the help of a standard learning algorithm using training data. The training data consists of signals with IQ samples of the received OFDM symbols along with the labels that indicate the modulation class of each OFDM signal. The trained model is then used to predict the modulation class of the received OFDM signal. In the following subsections, we provide the mathematical framework of DL for the task of AMC and describe the considered OFDM model.

A. OFDM SYSTEM MODEL

The discrete baseband OFDM samples $v_m[n]$ of mth OFDM symbol, generated by N-point inverse discrete Fourier transform (IDFT), can be written as [11]

$$v_m[n] = \sum_{k=0}^{N-1} V_m[k] e^{j2\pi kn/N}, \quad 0 \le n \le N-1 \tag{1}$$

where $N = \rho_s \times N_d$, ρ_s is the oversampling factor, N_d is the number of data subcarriers, $V_m[k]$ is the baseband modulated oversampled data obtained by zero-padding the baseband modulated information $\hat{V}_m[k]$. Thus, $V_m[k]$ is given by

$$V_m[k] = \begin{cases} \hat{V}_m[k] & 0 \le k \le N_d/2 - 1 \\ \hat{Z}_0 & N_d/2 \le k \le N_d(\rho_s - 1/2) - 1 \\ \hat{V}_m[k] & N_d(\rho_s - 1/2) \le k \le N - 1, \end{cases} \tag{2}$$

where $\hat{Z}_0$ is a vector of zeros of length $N_d(\rho_s - 1)$. To combat the effect of inter-symbol interference (ISI), CP of N_{cp} samples from the end of the OFDM symbol is added at the beginning of the OFDM symbol before the transmission. The transmitted baseband OFDM symbol $\bar{v}_m[n]$ of length $N+N_{cp}$, with CP is then given by

$$\bar{v}_m[n] = \begin{cases} v_m[n+N] & -N_{cp} \le n \le -1 \\ v_m[n] & 0 \le n \le N-1. \end{cases} \tag{3}$$

After passing through a frequency-selective fading channel with impulse response, $g[l]$ of length L, the received baseband OFDM samples of the mth OFDM symbol are given by

$$x_m[n] = e^{(j2\pi\epsilon n/N + \phi)} \sum_{l=0}^{L-1} g[l] \bar{v}_m[n-l-\tau] + \omega[n], \quad 0 \le n \le N_s - 1 \tag{4}$$

where ϵ is the normalized frequency offset, ϕ is the phase offset, τ is the timing offset, N_s is the length of the OFDM

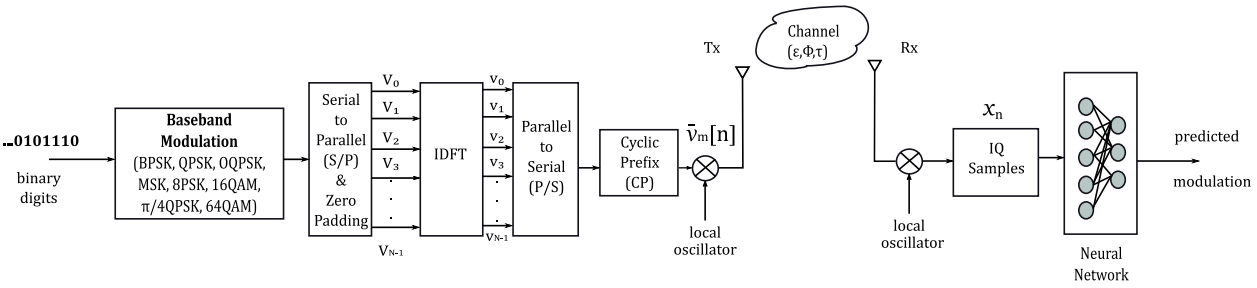


FIGURE 1. System model of the proposed DL-based AMC system for adaptive OFDM systems. The transmitter can adapt the modulation scheme or the number of subcarriers for the best end-to-end performance. The neural network is made up of CNN layers with residual connections.

symbol with CP, $N_s = N + N_{cp}$ and $N_{cp} \geq L$, and $\omega[n]$ is the additive white Gaussian noise (AWGN) with zero mean and variance σ_ω^2.

Collection of samples of $x_m[n]$ are denoted by x_n for $n = 1, 2, \ldots, M$ and received in serial, and samples from multiple such OFDM symbols are concatenated to obtain an ith instance (block) of input signal to the neural network. This is given by

$$\boldsymbol{x}^{(i)} = [x_1, x_2, \ldots, x_M]^T, \tag{5}$$

where M is the total number of samples or length of the input signal. The receiver has no knowledge of the OFDM symbol length, and the value of M is chosen independently of the length of the OFDM symbol N_s.

The set of all such training instances can be denoted by a matrix $\boldsymbol{X}$, in which the ith row $\boldsymbol{x}^{(i)}$, corresponds to one example consisting of IQ samples of the received OFDM signal. Therefore, the set of all training examples is given by

$$\boldsymbol{X} = [\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(K)}]^T, \tag{6}$$

where K is the total number of rows in $\boldsymbol{X}$, which is equal to the total number of training examples.

If the set of all modulation classes is denoted by $\Omega = \{\Omega_1, \Omega_2, \ldots, \Omega_C\}$, where C is the total number of modulation formats considered, then any given instance of the captured data, the signal belongs to one of these classes. This information about the modulation format of the given instance $\boldsymbol{x}^{(i)}$ is stored in the vector $\boldsymbol{y}^{(i)} = [y_1, y_2, \ldots, y_C]$, using one-hot encoding. Thus, if the given instance $\boldsymbol{x}^{(i)}$ belongs to a particular modulation format Ω_c, where $c = 1, 2, \ldots, C$, using one-hot encoding, we have

$$y_q = \begin{cases} 1 & \Omega_q = \Omega_c \\ 0 & \Omega_q \neq \Omega_c \end{cases} \quad \Omega_q \in \Omega. \tag{7}$$

The modulation formats of all K instances are stored in a matrix $\boldsymbol{Y}$, which is used during the training stage. Thus,

$$\boldsymbol{Y} = [\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \ldots, \boldsymbol{y}^{(K)}]^T, \tag{8}$$

where $\boldsymbol{y}^{(i)}$ represents the one-hot encoding of the modulation format of the signal $\boldsymbol{x}^{(i)}$.

After getting the K number of training examples and corresponding one-hot encoded modulation class, the dataset D is formed as

$$D = \left\{\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}\right\}_{i=1}^{K} \tag{9}$$

B. DL MATHEMATICAL FRAMEWORK

A DL model can be mathematically modeled as a composite non-linear mapping function

$$\hat{\boldsymbol{y}}^{(i)} = \mathcal{F}(\boldsymbol{x}^{(i)}; \boldsymbol{\Theta}), \tag{10}$$

where $\boldsymbol{x}^{(i)}$ is the input signal to the model, and $\boldsymbol{\Theta}$ is the set of parameters of the model that consists of weights and biases. DL uses computational models that consist of multiple processing layers to learn the representation of the input data with multiple levels of abstraction [37]. These multiple levels of abstraction are obtained by the composition of simple non-linear transformations at each layer. The processing layers are built with the interconnection of fundamental computational units called *neurons*. Multiple layers of the network are formed by the interconnection of neurons with the help of weights from one layer to another layer. These intermediate layers are also known as hidden layers. Such a multilayered network of neurons is commonly known as a deep neural network (DNN).

For a network with L layers, the composite mapping function $\mathcal{F}$ can be written as

$$\mathcal{F}(\boldsymbol{x}^{(i)}; \boldsymbol{\Theta}) = f_L(\ldots(f_2(f_1(\boldsymbol{x}^{(i)}; \boldsymbol{\theta}_1)); \boldsymbol{\theta}_2)\ldots; \boldsymbol{\theta}_L), \tag{11}$$

where f_l and $\boldsymbol{\theta}_l$ represents the non-linear transformation and parameters of lth hidden layer, respectively. For a convolution layer l with input h_l, the transformation f_l can be written in general as

$$f_l(\boldsymbol{h}_l, \boldsymbol{\theta}_l) = \sigma(h_l \otimes \boldsymbol{W}_l + \boldsymbol{b}_l), \qquad \boldsymbol{\theta}_l = [W_l, \boldsymbol{b}_l], \tag{12}$$

where $\otimes$ denotes the convolution operation, $\boldsymbol{h}_l$ is the output of layer $l-1$, σ is a non-linear mapping function commonly known as *activation* function, and W_l and $\boldsymbol{b}_l$ are parameters known as *weights* and *bias*, respectively.

The objective of DL is to learn the set of parameters $\boldsymbol{\Theta}$ that results in the best representation of input observation $\boldsymbol{x}^{(i)}$, for predicting the given output $\boldsymbol{y}^{(i)}$. The difference between the

predicted output $\hat{\boldsymbol{y}}^{(i)}$ and the actual output $\boldsymbol{y}^{(i)}$ is usually measured by an empirical loss function $\mathcal{L}(.)$. Thus, the objective is to train the model to learn the optimal set of parameters $\boldsymbol{\Theta}^*$ that minimizes the loss function over the whole set of training examples $\boldsymbol{X}$, given by

$$\boldsymbol{\Theta}^* = \arg\min_{\boldsymbol{\Theta}} \mathcal{L}(\mathcal{F}(\boldsymbol{X}; \boldsymbol{\Theta}), \boldsymbol{Y}). \tag{13}$$

For a multi-class classification with C different classes, the output of the model is a C dimensional vector where each element, $\hat{y}_c$, represents $p(\Omega_c|\boldsymbol{x}^{(i)}; \boldsymbol{\Theta})$, i.e., the probability that the given input $\boldsymbol{x}^{(i)}$ belongs to a particular class Ω_c. Therefore, $\mathcal{F}(\boldsymbol{x}^{(i)}; \boldsymbol{\Theta}) \in \mathbb{R}^C$ and can be written as

$$\begin{aligned}\hat{\boldsymbol{y}}^{(i)} &= \left[\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_C\right]^T \\ &= \left[p(\Omega_1|\boldsymbol{x}^{(i)}; \boldsymbol{\Theta}), p(\Omega_2|\boldsymbol{x}^{(i)}; \boldsymbol{\Theta}), \ldots, p(\Omega_C|\boldsymbol{x}^{(i)}; \boldsymbol{\Theta})\right]^T.\end{aligned} \tag{14}$$

Assuming that the set of all the received signals given by (6) are independent and identically distributed, the joint likelihood function is given by

$$\lambda(\boldsymbol{Y}|\boldsymbol{X}; \boldsymbol{\Theta}) = \prod_{i=1}^{K} \prod_{q=1}^{C} \left(p(\Omega_q|\boldsymbol{x}^{(i)}; \boldsymbol{\Theta})\right)^{y_q^{(i)}}. \tag{15}$$

Using (7), this can be further simplified as

$$\lambda(\boldsymbol{Y}|\boldsymbol{X}; \boldsymbol{\Theta}) = \prod_{i=1}^{K} \left(p(\Omega_c|\boldsymbol{x}^{(i)}; \boldsymbol{\Theta})\right)^{y_c^{(i)}}. \tag{16}$$

For large values of K, computation of the above expression might lead to underflow problems, and to avoid this, the logarithm of the likelihood function is usually considered. Taking the logarithm of (16), the log-likelihood is given by

$$\Lambda(\boldsymbol{Y}|\boldsymbol{X}; \boldsymbol{\Theta}) = \sum_{i=1}^{K} y_c^{(i)} \log p(\Omega_c|\boldsymbol{x}^{(i)}; \boldsymbol{\Theta}). \tag{17}$$

The optimal set of parameters $\boldsymbol{\Theta}^*$, can now be obtained by maximizing the log-likelihood function given by (17) as

$$\boldsymbol{\Theta}^* = \arg\max_{\boldsymbol{\Theta}} \Lambda(\boldsymbol{Y}|\boldsymbol{X}; \boldsymbol{\Theta}). \tag{18}$$

This is equivalent to minimizing the negative of the log-likelihood function. Therefore, (18) can be re-written as

$$\boldsymbol{\Theta}^* = \arg\min_{\boldsymbol{\Theta}} -\Lambda(\boldsymbol{Y}|\boldsymbol{X}; \boldsymbol{\Theta}). \tag{19}$$

Comparing (19) with (13), we have

$$\mathcal{L}(\mathcal{F}(\boldsymbol{X}; \boldsymbol{\Theta}), \boldsymbol{Y}) = -\Lambda(\boldsymbol{Y}|\boldsymbol{X}; \boldsymbol{\Theta}). \tag{20}$$

Substituting (17) in (20), the loss function is thus given by

$$\mathcal{L}(\mathcal{F}(\boldsymbol{X}; \boldsymbol{\Theta}), \boldsymbol{Y}) = -\sum_{i=1}^{K} y_c^{(i)} \log p(\Omega_c|\boldsymbol{x}^{(i)}; \boldsymbol{\Theta}). \tag{21}$$

Equation (21) is commonly known as categorical cross-entropy loss. The network is first initialized with parameters $\boldsymbol{\Theta}_0$ using any standard initialization procedure, and the optimal parameters $\boldsymbol{\Theta}^*$ are then computed iteratively using an optimization algorithm. By back-propagating the loss gradients across each layer of the network, the parameters $\boldsymbol{\Theta}$ are updated in each iteration until convergence.

$$\boldsymbol{\Theta}_{n+1} = \boldsymbol{\Theta}_n - \alpha \frac{\partial \mathcal{L}\left(\mathcal{F}(\boldsymbol{X}; \boldsymbol{\Theta}_n), \boldsymbol{Y}\right)}{\partial \boldsymbol{\Theta}_n}, \tag{22}$$

where α is a hyperparameter known as learning rate, which is used to control the rate of convergence.

III. NEURAL NETWORK ARCHITECTURE

For the task of AMC of OFDM signals, we use a special type of neural network called CNN, along with residual learning [35]. In the following subsections, we provide a brief description of CNNs and residual learning. The detailed architecture of the CNN used for the AMC of adaptive OFDM signals is discussed at the end of this section.

A. CONVOLUTIONAL NEURAL NETWORKS (CNNs)

CNNs are made up of a special type of layers that use convolution operations to extract useful representations from the input data. The non-linear mapping function of a CNN layer is given by (12). DL libraries make use of cross-correlation, a function similar to convolution. For a 1D input signal S, the cross-correlation function $Cor(i)$ is given by

$$Cor(i) = (Q \otimes S)(i) = \sum_{j} Q(j)S(i+j), \tag{23}$$

where Q is the convolution filter (also known as the kernel). The advantages of using CNN are *sparse connectivity*, *parameter sharing,* and *equivariance* to translation [38]. The equivariance property helps the network to be resilient to the translations in the time and frequency domain caused by the STO and CFO effects present in the received OFDM signal. Thus, CNNs help to overcome the impact of these impairments on classification performance. Sparse connectivity and parameter sharing result in less number of network parameters and hence reduces the storage requirement of the model. Sparse connectivity also results in fewer computations because of less number of connections, making it more feasible for the network to be deployed for real-time applications.

B. RESIDUAL LEARNING

It is often difficult to learn the underlying mapping function from input to output, especially when the training data consists of large variations. Such variations are common in the OFDM signal because of the non-linearity introduced by the hardware, impairments introduced from a frequency-selective fading channel that is time-varying, and different signal parameters introduced at the transmitters. With the residual learning approach, the network is trained to learn the residual mapping instead of the underlying mapping from input to output. This is realized by adding *shortcut* (also known as *skip*) connections from the input of a stacked layer to the output, as shown in Fig 2.

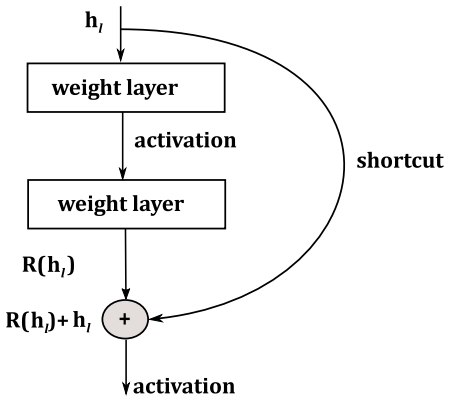


FIGURE 2. Residual block with stacked layers and shortcut connection.

Let us assume that a certain stacked layer l with input h_l learns a residual mapping function $R(h_l) = F(h_l) - h_l$, where $F(h_l)$ denotes the desired underlying non-linear mapping function. The output of the stacked layer after the skip connection is then given by $R(h_l)+h_l$. Thus, the stacked layer, along with skip connection, effectively learns the underlying non-linear mapping function $F(h_l) = R(h_l)+h_l$. This is illustrated in Fig 2. Residual connections are widely used for the application of denoising the images [36]. Similarly, by using the residual connections, the effect of random parameters such as CFO, STO, phase offset, and AWGN on the feature maps can be reduced. This leads to a better modulation classification performance.

C. NETWORK ARCHITECTURE

The CNN used for the AMC of OFDM signals in this work is designed based on the principles developed in [35], i.e., if the output feature map size remains the same, then the number of filters in the following layer also remains same. If the output feature map size is halved, then the number of filters in the following layer is doubled. Shortcut connections are then added to this network to form the final residual network used for the AMC of OFDM signals. The overall architecture of the network is shown in Fig. 3. The necessary hyperparameters to build the proposed architecture, as shown in Fig. 3, are tuned by using Bayesian optimization. The input to the network is a real two-dimensional signal containing IQ samples, obtained by separating the real and imaginary components of the complex baseband received OFDM signal. 1D convolutions are used in each layer, followed by batch normalization (BN). BN helps to reduce internal covariance shift and speeds up the training process [39]. Rectified linear unit (ReLU) is used as a non-linear activation function to introduce the non-linearity, thereby enabling the network to learn the complex relationship between the input and output. The output of the ReLU is given by

$$ReLU(\boldsymbol{W}_l\boldsymbol{h}_l + \boldsymbol{b}_l) = \max\{0, \boldsymbol{W}_l\boldsymbol{h}_l + \boldsymbol{b}_l\}. \quad (24)$$

Two types of shortcut connections are used for the residual blocks. When the dimension of the output of the stacked layer

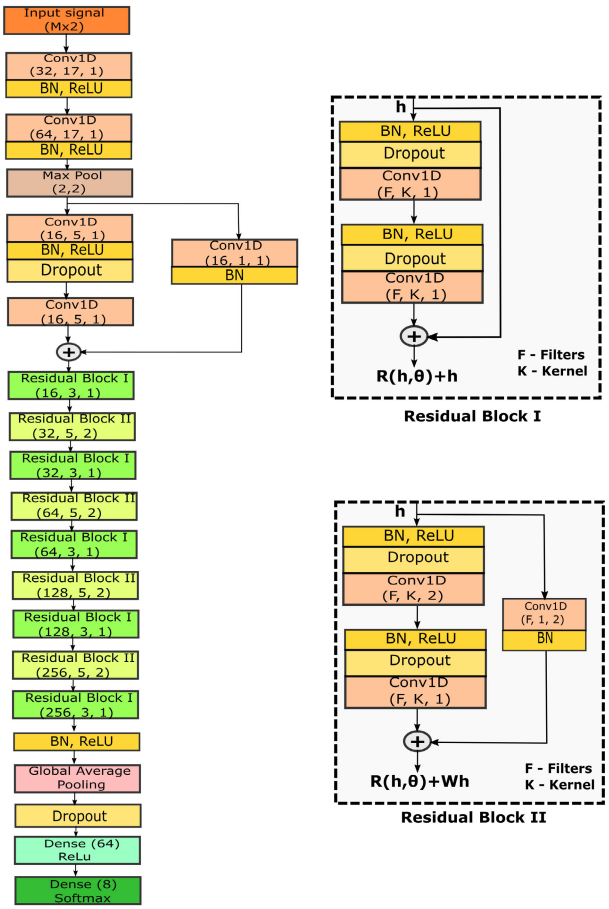


FIGURE 3. Deep neural network architecture used in the proposed work. The numbers in the brackets represent the number of filters, kernel size, and stride value, respectively.

is the same as its input, identity shortcut connections are used. The output from the lth residual block, in this case, is given by

$$h_{l+1} = R(h_l, \boldsymbol{\theta}_l) + h_l. \quad (25)$$

When the dimension of the input is changed because of the increased number of filters and convolutions with a stride value of 2, shortcut connection with linear projections W_l is used to match the dimensions. The linear projection is obtained by using a convolution layer with a filter size of 1. The output of such residual block can be written as

$$h_{l+1} = R(h_l, \boldsymbol{\theta}_l) + h_l \otimes W_l. \quad (26)$$

The output layer of the network is made up of dense connections with C neurons, where C is the number of modulation classes. Each neuron in the output layer represents a modulation format from the considered modulation classes, and the output of each neuron represents the probability that the given OFDM signal belongs to a particular modulation class. The one with the maximum probability value is chosen as the modulation format of the received OFDM signal. Since the output of this layer represents the probability distribution

as given by (14), the *softmax* activation function is used. If z_i represents the output of linear transformation from a neuron in the output layer, then the output $\hat{y}_i$ obtained after applying *softmax* activation is given by

$$\hat{y}_i = softmax(z_i) = \frac{\exp z_i}{\sum_{j=1}^{C} \exp z_j}. \tag{27}$$

The network parameters are initialized using the method proposed in [40], and Adam optimizer is used to train the model. The Adam optimizer combines the advantages of other optimization methods, such as AdaGrad and RMSProp and is found to perform well with noisy and sparse gradients [41]. Dropout [42] and L2 regularizer [43] are used in each layer as regularization techniques to prevent overfitting.

IV. EXPERIMENT AND RESULTS

We use Python framework with Keras [44] library using TensorFlow backend to build and train the neural network. MATLAB 2019a is used to generate the train and test dataset used for simulation. For the real-time experimental setup, we use NI USRP-2943R with GNU Radio platform. The model was trained using NVIDIA Quadro K1200 GPU.

A. DATASET GENERATION

The simulation data consists of IQ samples of the received baseband OFDM signal along with the modulation class label. The modulation classes considered are BPSK, QPSK, OQPSK, π/4-QPSK, MSK, 8-PSK, 16-QAM, and 64-QAM. For each of the modulations, examples were generated for a varying number of data subcarriers with $N_d = 16, 32, 64$, and 128 and with SNR values in the range of -10 dB to 20 dB in steps of 5 dB. Each modulation class consists of 256 examples for each of the subcarrier and SNR values. The data is split into 70% for training and 30% for validation. The test set consists of 256 examples per subcarrier and SNR value considered for each of the modulation classes. We use an oversampling factor of $\rho_s = 4$ and the cyclic-prefix of length $N_{cp} = 16$ samples. In order to create a more realistic scenario, we have generated Rayleigh and Rician channel coefficients using the standardized ITU-R [45] power delay profile, and specifically, it has been designed for outdoor-to-indoor and pedestrian test channel B. The distribution of STO, CFO, and phase offset are considered as shown in Table 1. In the proposed work, the context of time-varying channel is varied among the training examples along with varying the STO, CFO, and phase offsets uniformly between the values mentioned in Table 1.

The real-time data is generated in an indoor propagation environment using the NI-USRP setup, as shown in Fig.4. The setup contains a line-of-sight (LOS) path and hence can be considered to follow the Rician channel propagation. A sampling frequency of 2 MHz with a carrier frequency of 2.8 GHz was used for data transmission. The received downconverted baseband signal is stored as a long sequence of IQ samples in a data file for each modulation class. These long sequences are then sliced at random intervals to obtain the training and test data with the required number of samples. These data are then used to train and test the performance of the proposed network.

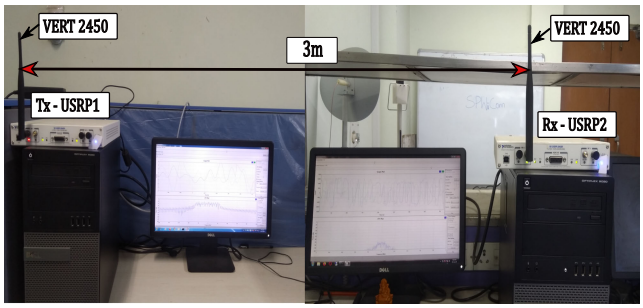


FIGURE 4. Real-time experimental laboratory setup using NI-USRP.

TABLE 1. Values of various parameters considered for simulation.

Parameter	Distribution
Fading environment	Rayleigh, Rician
K-factor (Rician)	10
Maximum Doppler Shift (f_D)	50Hz
STO (τ)	$\mathcal{U}[-8, 8]$
Normalized CFO (ϵ)	$\mathcal{U}[-0.4, 0.4]$
Phase offset (ϕ)	$\mathcal{U}[-\pi/2, \pi/2]$
CP length (N_{cp})	16
Number of subcarriers (N)	$64, 128, 256, 512$
Oversampling factor (ρ_s)	4

B. TRAINING

To achieve faster convergence, the training is performed in two stages. In the first stage, the model is trained with only data samples of SNR value 20 dB for 150 epochs. Adam optimizer is used with an initial learning rate of 0.001 for the first 40 epochs, and the learning rate is reduced to 0.0001 for the rest of the epochs. In the second stage, the model is initialized with the parameters learned in the first stage, and the training is continued further for 100 more epochs with the data samples of all SNR values. The value of batch size is set to 128, and a dropout value of 0.3 is used between the convolution layers. As the model's structure is independent of the number of samples contained in the input signal, the network is trained with signals of length $M = 4096$ samples and a various number of subcarriers. It is then tested for signals with various sample lengths of $M = 4096, 2048, 1024, 512$, and 256 samples. For the experimental setup, the model is trained with the input signals of length $M = 1024$ samples and then tested for the set of signals with the various number of samples considered.

C. SIMULATION RESULTS AND COMPARISON

The classification performance of the proposed AMC for the various number of samples per example (or signal length) is shown in Fig. 5. From the figure, it is evident that the classification accuracy improves with the increase in the number of samples of the received signal. This can be attributed to

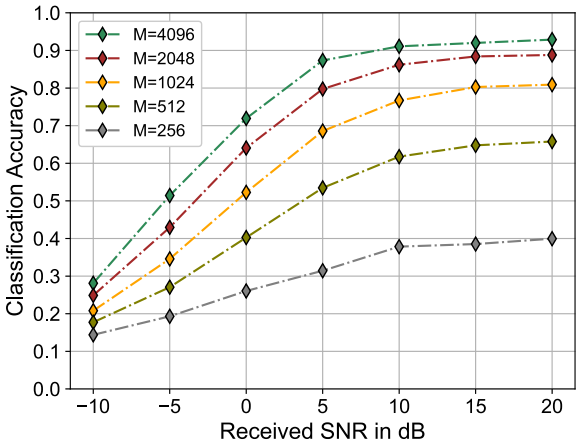


FIGURE 5. Classification accuracy vs. SNR for varying number of samples (M) per example.

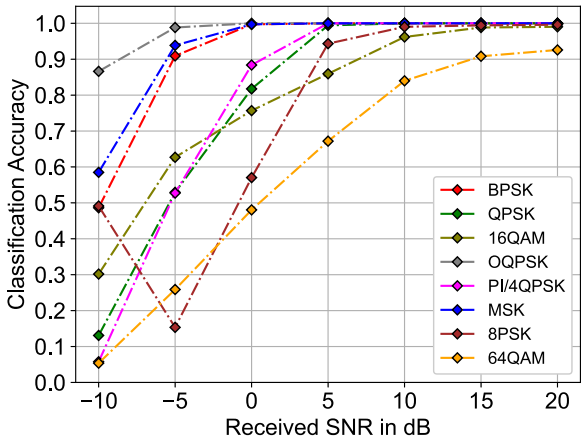


FIGURE 7. Classification accuracy vs. SNR for individual modulations considered (M=4096, N=64).

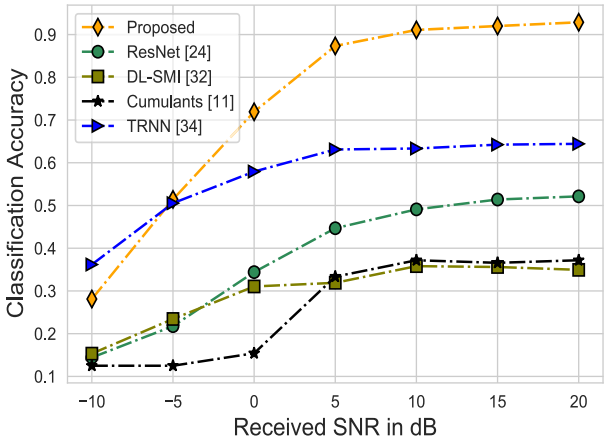


FIGURE 6. Classification performance comparison with the feature-based approach and other DL methods ($M = 4096$).

the larger number of OFDM symbols being present in the signals with more samples. Maximum classification accuracy of about 92% is obtained at a higher SNR value for signals with $M = 4096$ samples.

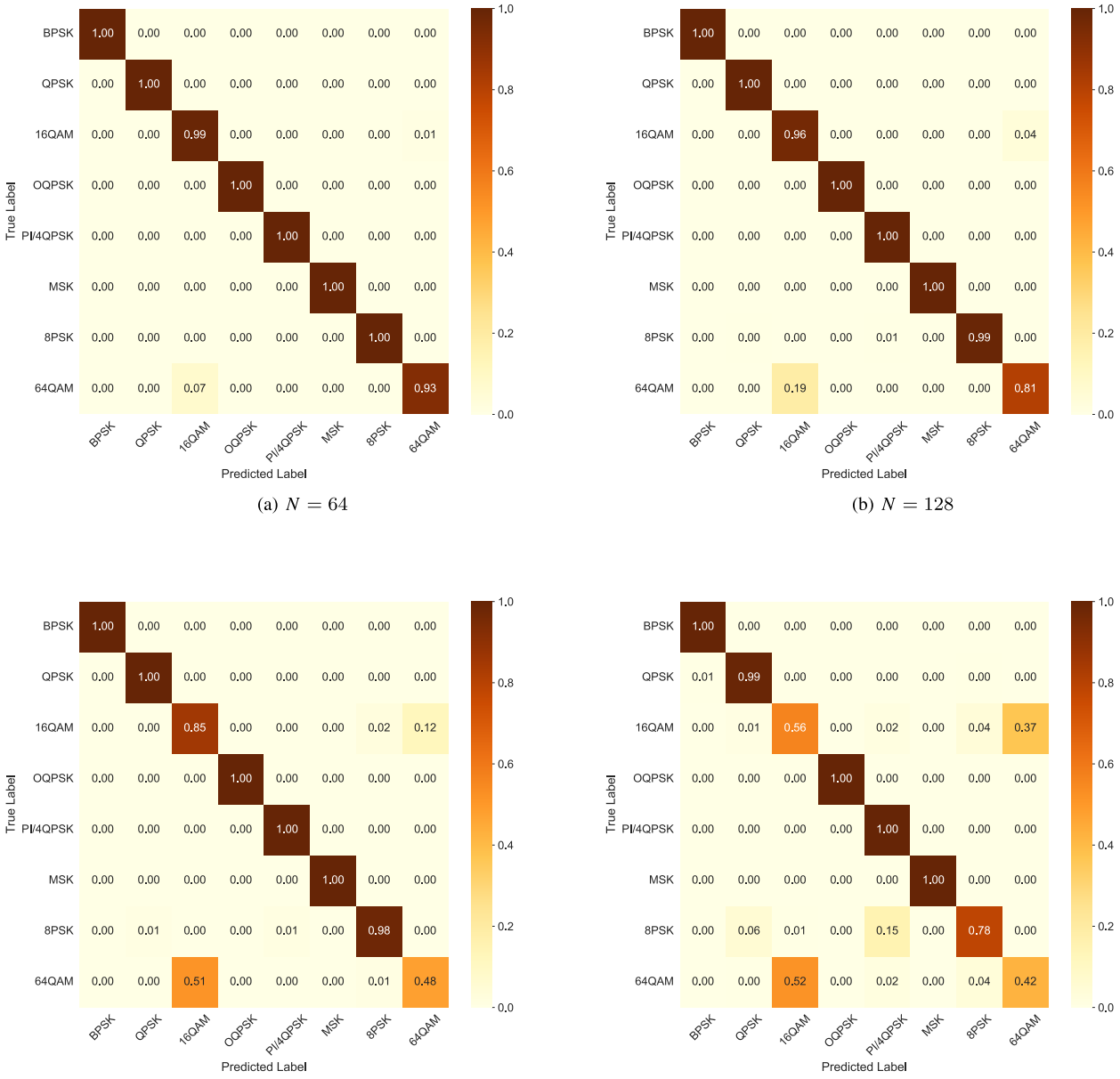
Fig. 6 presents the classification performance of the proposed CNN-based AMC, as compared to that of DL-signal modulation identification (DL-SMI) [32], triple-skip residual neural network (TRNN) [34], the feature-based approach [11] and ResNet [24] where DL-SMI, TRNN, and feature-based approach are for the AMC for OFDM systems and ResNet was proposed for the AMC of single carrier system. The proposed CNN-based AMC for OFDM is compared with a statistical approach that consists of features extracted using DFT and fourth-order cumulants presented in [11]. From the plot, it is evident that the proposed method outperforms the existing DL and statistical techniques. In the proposed model, the network has filters with a larger size in the initial layers, and this helps to capture the information present in the longer range of samples. The features presented in [11] fail to generalize well with added modulation classes, and hence, they exhibit poor classification performance for the considered set of modulation classes. The DL-SMI [32] method also fails to generalize with added modulations and under time-varying channel conditions. The TRNN [34] based method is limited in scope as it only looks at a specific subset of modulation techniques, i.e., BPSK, QPSK, and 8-PSK, and does not take into account how variations in signal parameters may impact the classification results.

The classification accuracies for individual modulations are shown in Fig. 7 for a subcarrier value of $N = 64$. All modulation classes reach over 90% accuracy by 10 dB SNR except 64-QAM. This is mainly due to the fact that the 16-QAM's constellation is a subset of the 64-QAM. Hence at the lower SNR values, 64-QAM is mostly confused for 16-QAM modulation because of the added noise. Fig. 8 presents the confusion matrix in terms of classification probabilities for the signals with $M = 4096$ samples at 20 dB SNR for the various number of subcarriers considered.

Fig. 9 shows the effect of the number of subcarriers on the classification performance for the signals with $M = 4096$ samples. The classification accuracy reduces with increasing number of subcarriers because as the number of subcarriers increases, the chances of interference between subcarriers also increases.

The effect of the propagation channel on the classification performance is shown in Fig. 10. The model performs better in the Rician channel in comparison with the Rayleigh channel. This can be attributed to the fact that the Rayleigh channel lacks a strong LoS component. Also, the model's performance is found to be slightly better with constant channel conditions having fixed CFO, STO, and phase offsets, as compared to that of time-varying channel conditions. The context of time-varying channel in the proposed work is that we are varying channels among the training example along with varying the STO, CFO, and phase offsets uniformly between the values mentioned in Table 1.

Fig. 11 shows the effect of variation of CFO, STO, and phase-offsets on the classification performance using the



(a) $N = 64$ (b) $N = 128$

(c) $N = 256$ (d) $N = 512$

FIGURE 8. Confusion matrix for various number of subcarriers considered at the received SNR value of 20 dB.

simulation data with an SNR of 10 dB. As mentioned in the figure, the value of CFO is varied from 0 to 0.4, while the magnitude of phase offset is varied from 0 to $\pi/2$, and similarly, STO is varied from 0 to 8 samples. To assess the effect of one impairment on the classification accuracy, the value of the other impairments was set to zero. For example, while assessing the model's performance for different ranges of variation of CFO, the values of STO and phase offset were set to 0. Also, in each case, the model was trained for the extreme value of impairment variation considered and then tested on datasets having different ranges of variation as shown in the plot. The maximum range of variation considered for each impairment is same as shown in Table 1. Results show that for a given impairment, the variation of classification accuracy is well within $\pm 2\%$. Also, for the dataset with only STO, the model exhibits lower classification accuracy in comparison to the datasets with CFO and phase offset.

D. EXPERIMENTAL RESULTS

Fig. 12 presents the classification performance of the proposed model, for various number of samples M, for the real-time captured signals. The network is trained with captured signals of length $M = 1024$ samples and tested with signals of various sample lengths. The system performs with

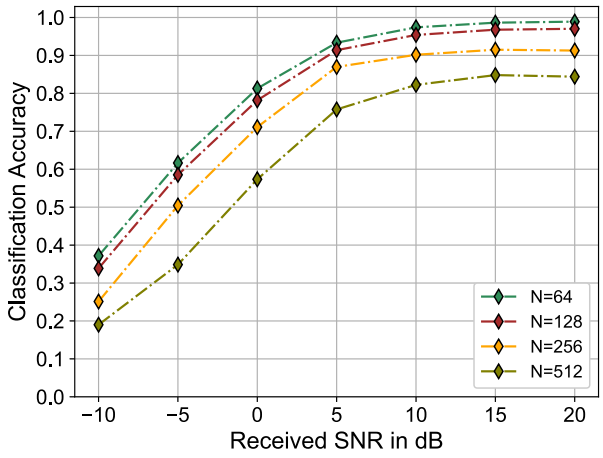


FIGURE 9. Classification accuracy vs. SNR for various number of subcarriers.

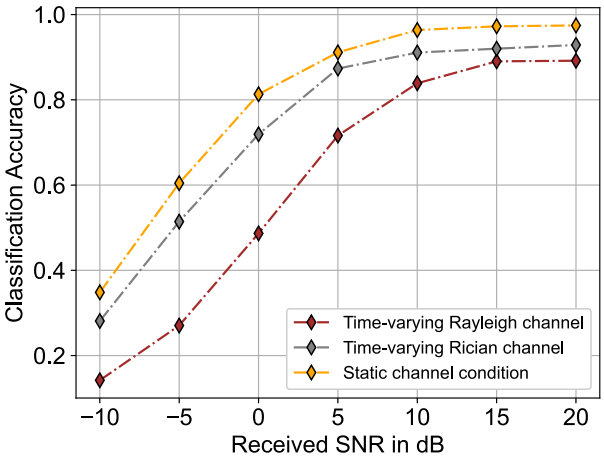


FIGURE 10. Classification accuracy vs. SNR plot for various channel conditions. In the case of a time-varying channel, the channel was considered to be varying from one example to another.

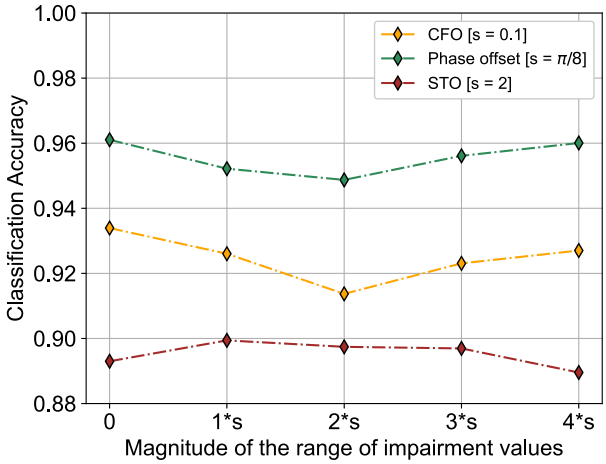


FIGURE 11. Variation of classification accuracy for different range of values of channel impairments considered. X-axis represents the magnitude of the maximum value of different impairments considered. The value of the received SNR was set 10 dB.

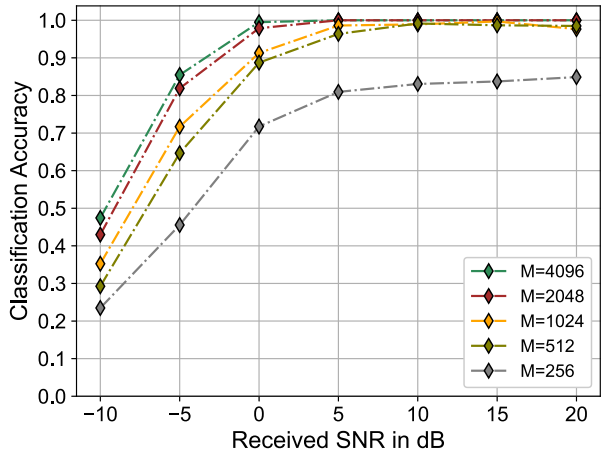


FIGURE 12. Classification accuracy vs. SNR plot for the real-time captured signals in lab environment for varying number of samples per example.

an accuracy value close to 100% for signals with $M = 4096, 2048, 1024$, and 512 samples at received SNR values of over 10 dB. The accuracy dropped for signals with $M = 256$ samples.

E. MODEL COMPLEXITY

The proposed model consists of 27 1D-convolutional layers and about 1.28M trainable parameters. The model is built, trained, and tested on a system equipped with Intel Xeon E5 CPU and NVIDIA Quadro K1200 GPU. The training process takes about 14 hours to converge. Once the model is trained, it takes approximately 110 ms to predict the modulation class of a given input signal with $M = 4096$ samples which is lower than DL-SMI [32] and slightly higher than ResNet [24] and TRNN [34], as shown in Table 2. Although the proposed model is slightly more time complex than ResNet [24] and TRNN [34], but outperforms in terms of classification accuracy.

The computational complexity of existing and proposed model is given in Table 2. where K_1 denotes the size of filter, n_1 denotes the length of input, d_1 denotes the depth of input, F_1 denotes the number of filters, $c_{l_1} \times e_{l_1}$ is input feature map at the lth dense layer, q denotes the number of convolution layer and v is the number of hidden dense layers in the proposed model. The notation with a subscript (2,3,4) in the complexity term of the DL-SMI [32], ResNet [24], and TRNN [34] methods has the same meaning as described for the proposed method, respectively. As $v_2 > v_1$ and $c_{l_2} > c_{l_1}$ hence DL-SMI [32] model is more computationally complex than proposed model. ResNet [24] contain $q_3 = 32(> q_1)$ convolutional layers and 128, 128 hidden neurons in the 2 $(v_3 > v_1)$ hidden layers which makes more computationally complex than the proposed method. As in TRNN [34] $q_4 = 42(> q_1)$ which makes it more computationally complex than proposed method.

TABLE 2. Comparison of Computational Complexity.

Network	Complexity ($\mathcal{O}$)	Parameters	Prediction time (ms/example)
Proposed Model	$\mathcal{O}\{\max(q_1K_1n_1d_1F_1, v_1c_{l_1}e_{l_1})\}$	$1,287,160$	110
DL-SMI [32]	$\mathcal{O}\{\max(q_2K_2n_2d_2F_2, v_2c_{l_2}e_{l_2})\}$	$135,958,472$	260
ResNet [24]	$\mathcal{O}\{\max(q_3K_3n_3d_3F_3, v_3c_{l_3}e_{l_3})\}$	$237,034$	105
TRNN [34]	$\mathcal{O}\{\max(q_4K_4n_4d_4F_4, v_4c_{l_4}c_{l_4})\}$	$433,384$	79

The number of trainable parameters of the proposed model and existing DL-based AMC technique is given in Table 2. The proposed method for modulation classification has been shown to be effective in terms of classification accuracy. Specifically, the method has improved accuracy by 28% and 40%, respectively, compared to the ResNet [24] and TRNN models [34], despite having a slightly larger number of parameters. Additionally, compared to the DL-SMI model [32], the proposed method has reduced the number of parameters by 1.34M while still improving accuracy by 56%. These results demonstrate the effectiveness of the proposed method in improving modulation classification performance.

V. CONCLUSION

In this paper, we have proposed an AMC technique for adaptive OFDM systems using CNN with residual learning from received baseband IQ samples. The main advantage of the proposed method over the traditional and statistical feature-based methods is that the proposed method is completely data-driven and assumes no prior knowledge about the signal or channel statistics for a varying number of subcarriers. Compared to the existing DL method, the proposed method can be used to classify signals in the presence of varying number of subcarrier, CFO, STO, and phase offset with unknown channel conditions. Obtained results show that the proposed method outperforms the traditional feature-based approaches and is able to classify a broad range of modulation classes, i.e., BPSK, QPSK, OQPSK, $\pi/4$-QPSK, MSK, 8-PSK, 16-QAM, and 64-QAM. The validity of the proposed method is verified in real-time using the NI-USRP setup, and the results obtained are in good agreement with the simulation. The one limitation of DL-based AMC is that the model needs to be trained if the new set of modulations adopts in the system.

REFERENCES

[1] S. Majhi, M. Kumar, and W. Xiang, "Implementation and measurement of blind wireless receiver for single carrier systems," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 8, pp. 1965–1975, Aug. 2017.

[2] S. Majhi and T. S. Ho, "Blind symbol-rate estimation and test bed implementation of linearly modulated signals," *IEEE Trans. Veh. Technol.*, vol. 64, no. 3, pp. 954–963, Mar. 2015.

[3] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

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

[5] A. K. Pathy, A. Kumar, R. Gupta, S. Kumar, and S. Majhi, "Design and implementation of blind modulation classification for asynchronous MIMO-OFDM system," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021.

[6] M. L. D. Wong and A. K. Nandi, "Semi-blind algorithms for automatic classification of digital modulation schemes," *Digit. Signal Process.*, vol. 18, no. 2, pp. 209–227, Mar. 2008.

[7] S.-Z. Hsue and S. S. Soliman, "Automatic modulation classification using zero crossing," *IEE Proc. F Radar Signal Process.*, vol. 137, no. 6, pp. 459–464, Dec. 1990.

[8] A. Swami and B. M. Sadler, "Hierarchical digital modulation classification using cumulants," *IEEE Trans. Commun.*, vol. 48, no. 3, pp. 416–429, Mar. 2000.

[9] M. Pedzisz and A. Mansour, "Automatic modulation recognition of MPSK signals using constellation rotation and its 4th order cumulant," *Digit. Signal Process.*, vol. 15, no. 3, pp. 295–304, May 2005.

[10] R. Gupta, S. Kumar, and S. Majhi, "Blind modulation classification for OFDM in the presence of timing, frequency, and phase offsets," in *Proc. IEEE 90th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2019, pp. 1–5.

[11] R. Gupta, S. Kumar, and S. Majhi, "Blind modulation classification for asynchronous OFDM systems over unknown signal parameters and channel statistics," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5281–5292, May 2020.

[12] R. Gupta, S. Majhi, and O. A. Dobre, "Blind modulation classification of different variants of QPSK and 8-PSK for multiple-antenna systems with transmission impairments," in *Proc. IEEE 88th Veh. Technol. Conf. (VTC-Fall)*, Aug. 2018, pp. 1–5.

[13] S. Majhi, R. Gupta, W. Xiang, and S. Glisic, "Hierarchical hypothesis and feature-based blind modulation classification for linearly modulated signals," *IEEE Trans. Veh. Technol.*, vol. 66, no. 12, pp. 11057–11069, Dec. 2017.

[14] R. Gupta, S. Majhi, and O. A. Dobre, "Design and implementation of a tree-based blind modulation classification algorithm for multiple-antenna systems," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 8, pp. 3020–3031, Aug. 2019.

[15] C.-S. Park, J.-H. Choi, S.-P. Nah, W. Jang, and D. Y. Kim, "Automatic modulation recognition of digital signals using wavelet features and SVM," in *Proc. 10th Int. Conf. Adv. Commun. Technol.*, Feb. 2008, pp. 387–390.

[16] K. M. Ho, C. Vaz, and D. G. Daut, "A wavelet-based method for classification of binary digitally modulated signals," in *Proc. IEEE Sarnoff Symp.*, Mar. 2009, pp. 1–5.

[17] D. Avci, "An intelligent system using adaptive wavelet entropy for automatic analog modulation identification," *Digit. Signal Process.*, vol. 20, no. 4, pp. 1196–1206, Jul. 2010.

[18] E. Akleman, "Deep learning," *Computer*, vol. 53, no. 9, p. 17, Sep. 2020.

[19] H. He, C. Wen, S. Jin, and G. Y. Li, "Model-driven deep learning for MIMO detection," *IEEE Trans. Signal Process.*, vol. 68, pp. 1702–1715, 2020.

[20] J. Ma, H. Liu, C. Peng, and T. Qiu, "Unauthorized broadcasting identification: A deep LSTM recurrent learning approach," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 9, pp. 5981–5983, Sep. 2020.

[21] H. Xie, Z. Qin, G. Y. Li, and B. Juang, "Deep learning enabled semantic communication systems," *IEEE Trans. Signal Process.*, vol. 69, pp. 2663–2675, 2021.

[22] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cognit. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.

[23] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Proc. Int. Conf. Eng. Appl. neural Netw.* Cham, Switzerland: Springer, 2016, pp. 213–226.

[24] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 168–179, Feb. 2018.

[25] Y. Wang, M. Liu, J. Yang, and G. Gui, "Data-driven deep learning for automatic modulation recognition in cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 4074–4077, Apr. 2019.

[26] G. J. Mendis, J. Wei, and A. Madanayake, "Deep learning-based automated modulation classification for cognitive radio," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Dec. 2016, pp. 1–6.

[27] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors," *IEEE Trans. Cognit. Commun. Netw.*, vol. 4, no. 3, pp. 433–445, Sep. 2018.
[28] S. Peng, H. Jiang, H. Wang, H. Alwageed, and Y. Yao, "Modulation classification using convolutional neural network based deep learning model," in *Proc. 26th Wireless Opt. Commun. Conf. (WOCC)*, Apr. 2017, pp. 1–5.
[29] S. Peng, H. Jiang, H. Wang, H. Alwageed, Y. Zhou, M. M. Sebdani, and Y. Yao, "Modulation classification based on signal constellation diagrams and deep learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 718–727, Mar. 2019.
[30] Y. Xu, D. Li, Z. Wang, Q. Guo, and W. Xiang, "A deep learning method based on convolutional neural network for automatic modulation classification of wireless signals," *Wireless Netw.*, vol. 25, no. 7, pp. 3735–3746, Oct. 2019.
[31] F. Shi, Z. Hu, C. Yue, and Z. Shen, "Combining neural networks for modulation recognition," *Digit. Signal Process.*, vol. 120, Jan. 2022, Art. no. 103264.
[32] S. Hong, Y. Zhang, Y. Wang, H. Gu, G. Gui, and H. Sari, "Deep learning-based signal modulation identification in OFDM systems," *IEEE Access*, vol. 7, pp. 114631–114638, 2019.
[33] J. Shi, S. Hong, C. Cai, Y. Wang, H. Huang, and G. Gui, "Deep learning-based automatic modulation recognition method in the presence of phase offset," *IEEE Access*, vol. 8, pp. 42841–42847, 2020.
[34] L. Zhang, C. Lin, W. Yan, Q. Ling, and Y. Wang, "Real-time OFDM signal modulation classification based on deep learning and software-defined radio," *IEEE Commun. Lett.*, vol. 25, no. 9, pp. 2988–2992, Sep. 2021.
[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[36] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.
[37] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[38] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org
[39] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, vol. 37, Jul. 2015, pp. 448–456.
[40] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.
[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[42] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 56, pp. 1929–1958, 2014.
[43] A. Y. Ng, "Feature selection, L_1 vs. L_2 regularization, and rotational invariance," in *Proc. 21st Int. Conf. Mach. Learn. (ICML)*, 2004, p. 78.
[44] F. Chollet. (2015). *Keras*. [Online]. Available: https://keras.io
[45] *Guidelines for Evaluation of Radio Transmissions Technologies for IMT-2000X*, document ITU-RM.1225, Geneva, Switzerland, Dec. 2009.

ANAND KUMAR (Graduate Student Member, IEEE) received the B.Tech. degree in electronics and communication from Dr. A. P. J. Abdul Kalam Technical University, Uttar Pradesh, India, in 2015, and the M.Tech. degree in telecommunication engineering from NIT, Durgapur, India, in 2018. He is currently a Research Scholar with the Department of Electrical Engineering, IIT Patna, India. His research interests include signal processing for wireless communication, which includes modulation classification, OFDM, MIMO, MIMO-OFDM, IRS, and OTFS.

KEERTHI KUMAR SRINIVAS received the B.E. degree in electronics and communication from PESIT, South Campus, Bengaluru, India, in 2014, and the M.Tech. degree in communication system engineering from IIT Patna, India, in 2020. He is currently associated with Altiostar, A Rakuten Symphony Company, Bengaluru. His research interests include digital signal processing, communication theory, wireless communications, machine learning, deep learning and its application to PHY layer of wireless communications.

SUDHAN MAJHI (Senior Member, IEEE) received the M.Tech. degree in computer science and data processing from the Indian Institute of Technology Kharagpur, Kharagpur, India, in 2004, and the Ph.D. degree from Nanyang Technological University (NTU), Singapore, in 2008. He was a Postdoctoral Researcher with the University of Michigan-Dearborn, Dearborn, MI, USA, the Institute of Electronics and Telecommunications, Rennes, France, and NTU. He is currently an Associate Professor with the Department of Electrical Communication Engineering, Indian Institute of Science (IISc), Bengaluru, India. His research interests include signal processing for wireless communication, which includes blind synchronization and parameter estimation, modulation classification, cooperative communications, physical layer security, cognitive radio, OFDM, SC-FDMA, MIMO, MIMO-OFDM NOMA, mmWAVE, OTFS, IRS, and sequence design.