Electronics and Electrical Engineering

https://www.aimspress.com/journal/ElectrEng

AIMS Electronics and Electrical Engineering, 10(3): 504–526
DOI: 10.3934/electreng.2026020
Received: 09 December 2025
Revised: 17 May 2026
Accepted: 03 June 2026
Published: 02 July 2026

Research article

A novel D-CNL-R classifier approach for automatic modulation classification

Tamizhelakkiya K* **and C.T. Manimegalai**

Department of Electronics and Communication Engineering, SRM Institute of Science and Technology, Chennai, Tamil Nadu, 603203, India

* **Correspondence:** Email: tk1045@srmist.edu.in, tamizhjai@gmail.com.

Abstract: Automatic Modulation Classification (AMC) is a significant decision-making process in non-cooperative, 5G, and beyond communication systems. Advancements in Artificial Intelligence (AI) led to the implementation of Deep Learning (DL) to provide superior performance over the feature extraction and offline training process of AMC. In this work, we proposed a hybrid modulation classification architecture by integrating Convolutional Neural Networks (CNN) with ML classifiers such as Random Forest (RF), Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost). The Radio Frequency Signal Classification (RFSC) dataset consists of data collected under different Signal-to-Noise Ratio (SNR) scenarios to analyze the resilience of the classifiers. Among all architectures, the Deep Convolutional Layer based RF (D-CNL-R) model achieved superior modulation recognition accuracy due to its enhanced capability to learn complex nonlinear feature distributions. We observed that the training overhead of the proposed D-CNL-R reduced to $\simeq 1\times$ with better classification accuracy performance. We also presented an experimental approach for the prediction performance of real-time signals for indoor and outdoor scenarios.

Keywords: deep learning; modulation recognition, CNN, machine learning, SNR

1. Introduction

The future 5G and beyond cellular systems [1–4] are expected to support a billion users with a trillion devices, namely massive Machine-Type Communications (mMTC), and Internet-of-Vehicles (IoV) services. With the high cost of signaling overhead and energy consumption, massive devices place a huge traffic load on base stations. In [5], Automatic Modulation Classification (AMC) can be adopted to blindly identify modulation classes without handshaking, thus reducing the signaling overhead of physical channels. Thus, AMC plays a vital role in future Artificial Intelligence (AI)-based intelligent modems to enhance resource allocation and dynamic scheduling. Moreover, it has been

widely used in electronic warfare to determine the enemy's intercepted communications in military environments [6]. Furthermore, it has been extended to spectrum monitoring [7], Link Adaptation (LA) [8], Dynamic Spectrum Access (DSA) [9], traffic prediction [10], and Cognitive Radio (CR) [11] to allow better spectrum efficiency and higher data rates. Traditionally, AMC has also been considered as a conventional pattern recognition problem in the field of feature extraction and classification.

Machine Learning (ML)-based approaches applied in the areas of multi-class Radio Frequency Interference (RFI) detection [12] and AMC [13]. However, in classical ML approaches, the complexity of activities such as data preprocessing, feature extraction, feature selection, etc., severely reduces the precision in terms of both efficiency and accuracy.

Deep Learning (DL) is a sub-class of ML, have shown exceptional results in the context of AMC [14] to address the challenges mentioned above. The enormous amount of data processing layers in a hierarchical design is used in DL-based algorithms for pattern recognition and feature extraction [15]. Convolutional Neural Networks (CNN) have been utilized for object detection in computer vision domains [16]. In addition, DL [17] lacks a sufficient dataset to train the model, particularly in supervised learning applications. To solve this problem, image-based CNNs have become more popular in recent days. It is a process of reusing pre-defined CNN models on a huge dataset that has already been built, such as the ImageNet project [18].

1.1. Related works

In a wireless communication system, AMC is a key approach to accurately detect and decode signals. A two-step procedure consists of signal pre-processing [19] and model development phases. The major advancements in the field of ML have been achieved by improving the learning algorithm, hardware resources, and training datasets [20]. Training datasets are limited in many important AI applications, such as Natural Language Processing (NLP), computer vision, and modulation recognition [21]. The Cycle Frequency-Domain Profile (CDP) detection technique has been used to extract the necessary features [22], which are fed into neural networks for a classification task [23]. To improve the efficiency of AMC algorithms, Multi-Layer Perceptron (MLP) [24], CNN [25], Residual Network (ResNet), and Dense Convolutional Network (DenseNet) have been developed.

Traditional CNN models such as Alexnet [26] and GoogleNet [27] achieve higher classification accuracy at the cost of significant computational complexity. Additionally, the LeNet [28] architecture has been tested for various 3GPP standard waveforms such as $\frac{\pi}{2}$-BPSK, QPSK, 16QAM, 64QAM and 256QAM. In addition, the CNN model has been evaluated for four modulation classes, such as BPSK, QPSK, 8PSK, and 16QAM [29]. In related work, it has been found that predominantly I/Q-based samples are adopted for classification. In [30], a novel accumulated polar feature-based AMC has been demonstrated for both online and offline training processes along with channel compensation techniques. Recently, Recurrent Neural Network (RNN) models [31], such as Long Short-Term Memory (LSTM) [32], Gated Recurrent Unit (GRU) [33], and Convolutional Long Short-Term Memory Deep Neural Network (CLDNN) [34] have been developed for modulation classification that adopts temporal-related features. Hybrid frameworks have been explored, such as combining a homogeneous ensemble of identical Deep Neural Network (DNN) with a heterogeneous hybrid architecture [35] and hand-crafted features with Support Vector Machines (SVMs) [36].

1.2. Problem statement

From the literature review, it has been found that there are still difficult problems in modulation classification tasks. In addition, the performance of the classifier is entirely dependent on the quality of these manually engineered features, which have high computational complexity. However, traditional ML methods are based on prior information and variations in the sensitivity of the system. Complex problems might require more features to achieve acceptable classification accuracy. However, the CNN model has achieved better results by mapping informative features into images, but its classification performance for real-time signals is still in an early development stage. Therefore, the classification process becomes an essential part of the AMC process and hence gains more attention towards the adoption of classifiers. The ensemble learning process used in the RF classifier achieves better accuracy through the principle of majority voting. Such a generalized process finds it more attractive for complex RFSC datasets and expects further improvements in the classification accuracy performance.

Primarily, the process of converting one-dimensional (1D) time-domain data into a two-dimensional (2D) image dataset using unsigned integer (uint8) encoding technology can be performed without noise-suppression preprocessing and retains valuable information in the original signal. The feature extraction strategies require complex models for feature detection, which leads to the design of low protractible futuristic model. Hence, feature detection becomes a requirement to overcome the lack of informative features. Secondly, DL methods are not precisely discussed from the point of view of AMC. Finally, the capabilities of the DL model's feature extraction and the ML as a classifier basis have been stacked to improve the overall classification accuracy. To address these issues and implement models with higher classification performance, we propose a Deep Convolutional Layer based ML classifier (D-CNL-M) model.

1.3. Contributions

The core objective is to apply the proposed D-CNL-M model to AMC using the RFSC dataset. The RFSC dataset has been created using a transmitter (Deep Radio) and a Receiver (Wi-Guy). The dataset consists of seven modulation classes (CPFSK, GFSK, 64QAM, BPSK, 16QAM, GMSK, and QPSK) whose samples are captured via software defined radio (SDR) in real-time.

- The proposed D-CNL-M model has been applicable to any image-based classification problem since it can learn the features from any image dataset. Therefore, we can train a reasonable count of convolutional layers on the dataset and use their predictions as input features to the ML-classifier that reinforces the reliability of the final prediction output. The proposed model combines the classification ability of ML with the feature extraction of convolutional layers.
- A research methodology to implement the proposed D-CNL-M model for training, testing, and recognition of modulation classes has been elaborated in detail. A comprehensive analysis and performance evaluation of the baseline CNN model has been compared with the proposed hybrid models such as Deep Convolutional Layer based RF classifier (D-CNL-R), Deep Convolutional Layer based Support Vector Machine classifier (D-CNL-S) and Deep Convolutional Layer based Extreme Gradient Boosting (D-CNL-X). The offline training overhead has been evaluated and presented for different models.
- The performance metrics of different hybrid models with CNN model have also been analyzed. We also extend the work to identify the modulation type used in real-time cellular signal

transmissions, which are captured from nearby cell towers under different scenarios.

2. System model

Deep Radio (DR) combines the GNU companion, a telescopic antenna, and HackRF to form a transceiver. HackRF is a popular SDR device that can both send and receive signals over a wide range of 50 MHz to 6 GHz. The in-built open-source GNU platform can be programmed and managed as a standalone system. The Wi-Guy receiver comprises RTL-SDR to capture the samples from the transmitter. The received signal $r(t)$ samples are decimated and fed into the RF to matrix conversion block, which converts RF samples into image format with size ($I_H \times I_W \times I_C$), where I_H is the height size, I_W is the width, I_C is the number of channels or nature of the image, respectively. It is fed as input to the CNN model, which performs feature extraction, providing details on low and high-level features. These features are passed into the NN/ML classifier to perform modulation classification. The modulation class has been identified in real-time cellular signals to provide the best decision strategy to maximize performance without causing any interference in future cellular systems.

2.1. Signal model for data acquisition and pre-processing

We have chosen two SDRs for transmission [37], and reception [38] of different modulated waveforms using DR and Wi-Guy [39] respectively. The received baseband complex signal envelope can be written as

$$r(t) = s(t; m_i) + g(t) \tag{2.1}$$

where, $s(t; m_i)$ defines the complex envelope of the received signal, and $g(t)$ denotes complex Additive White Gaussian Noise (AWGN). The noise-free signal $s(t; m_i)$ can be modeled as [40],

$$s(t; m_i) = \alpha e^{j2\pi(\Delta f t+\theta)} \sum_{j=0}^{l} e^{j\phi_j} s_j^i u(t - (j-1)T_s - \epsilon T_s) \tag{2.2}$$

where, α denotes the amplitude of the signal, Δf represents the carrier frequency offset, θ is the carrier phase, ϕ_j is phase jitter, T_s is the symbol period, ϵ is the time epoch between the transmitter and receiver, $u(t) = p(t) \otimes h(t)$ defines the nature of channel response $h(t)$ with transmit pulse shape $p(t)$. m_i represents a channel vector for the i^{th} modulation class. The main objective of the work is to identify the modulation class from the captured RF signal $r(t)$ samples. The raw I/Q samples are converted into dataset vectors (using uint8 encoding technology) and further converted into a suitable image format ($I_H \times I_W \times I_C$) that is the most suitable input size for Convolution layers [41]. The advantage is that it explores the 2D features of the original signal by retaining the original informative features in the data without expert assistance. In our work, for each raw 1D signal consisting of 1568 data samples, the gray matrix image size has been set to $28 \times 28 \times 2$.

2.2. Proposed model

The convolutional layer extracts the signal features with the dimensions 3×3 of the convolutional kernels and the max pooling operations as shown in Figure 1. Deep convolutional layers have been applied to extract more representative features from raw I/Q data with a higher degree of dimension

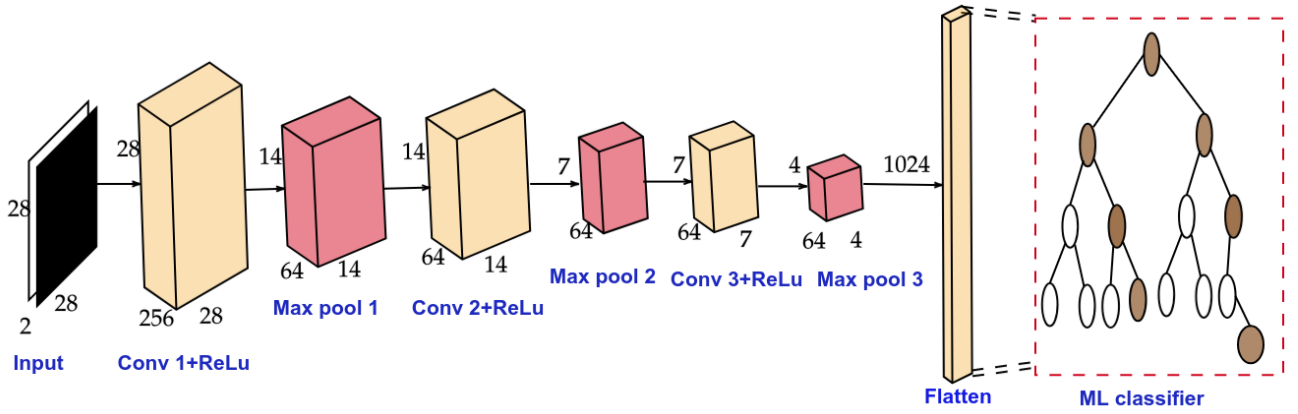


Figure 1. The Proposed D-CNL-M Architecture for AMC: Three convolutional layers and a flattened layer for feature extraction; ML is a classifier.

while choosing the training samples. Moreover, the adoption of convolution filters enables feature extractors in the convolutional layers to be tolerant of shifting the noise while training the data samples. The zero-padding has been adopted to optimize the structure of the convolutional layers to control overfitting issues. The ML-classifier has a stronger generalization ability for classifying the various modulation types than the NN classifier.

The ML-classifier is more complex and reliable than the NN classifier, since it is composed of boost and bagging strategies to obtain better classification performance. The proposed D-CNL-M model integrates the capabilities of the convolutional layer as a feature extractor, and the ML model works as a classifier. The ML-classifier performs better on input features represented in tabular format than on raw samples. However, in case the NN classifier can be formed by a network of neurons that cannot operate without each other and is grouped into layers for data processing and move towards the upcoming layers. The neurons available in the last layer play an major role in making the decision.

2.3. Classification and decision

The model can be tuned to extract the features from the series of convolutional layers that have been used as an input to the NN/ML classifier.

- **Random Forest (RF) classifier:** The R-classifier is a type of ML technique that consists of many Decision Trees (DTs), that generates the DTs $(D_1, D_2, ..., D_n)$ for "n" times of single datasets. For each D_i, a complete decision tree $t_i()$ is trained from a random selection of "k" features of dataset that have totally "m" features, where $k < m$. The algorithm will select the root node with the maximum information gained. Then, it divides into child nodes and continues "n" times to form a RF with "n" trees. Every tree relies on an independent random sample to make a decision, and the most popular class appears as the classification result, as shown in Figure 2. The RF classifier [42] is defined as

$$t(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} t_i(\mathbf{x}) \tag{2.3}$$

where, $\mathbf{x}$ defines the label of each modulation class. The classification of input $\mathbf{x}$ starts from the root node, and the above-mentioned process has been repeated until it reaches the corresponding

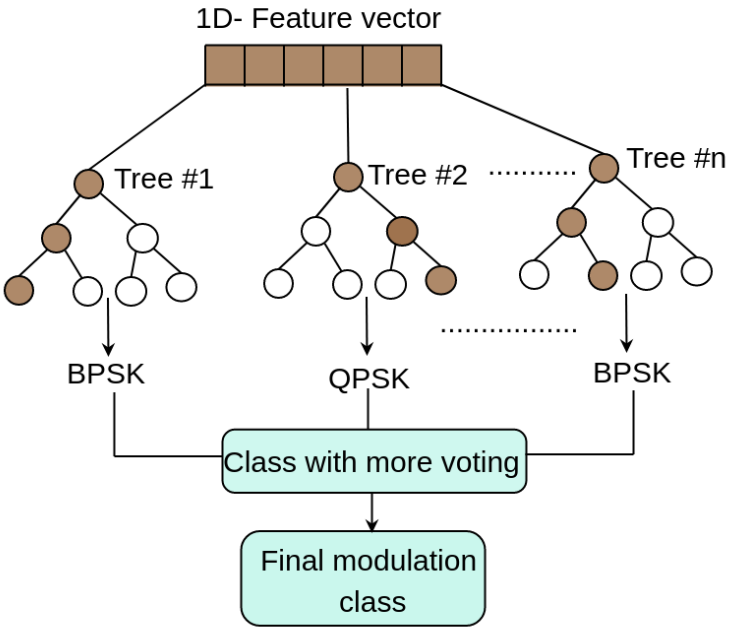


Figure 2. Schematic of a RF classifier with n DTs. The final decision on classification is determined by majority voting by individual DT.

modulation class. The NN uses the softmax activation function, which estimates the maximum probability between different modulation classes. The R-classifier has been configured with 50 decision trees including parameters such as Gini impurity as the splitting criterion, square-root feature sampling, unrestricted tree depth, and bootstrap sampling.

- **Support Vector Machine (SVM) classifier:** This is a type of ML classifier that determines an optimal path to maximize the separation margin between different modulation classes. Given an input feature vector $\mathbf{x}$, the decision boundary of the SVM classifier is represented as

$$\mathbf{w}^T\mathbf{x} + b = 0 \tag{2.4}$$

where $\mathbf{w}$ and b denote the weight vector and bias term, respectively. The SVM optimization objective is formulated as

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 \tag{2.5}$$

subject to

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \tag{2.6}$$

where y_i indicates the class label corresponding to the input feature vector $\mathbf{x}_i$. In this work, the Radial Basis Function (RBF) kernel is employed to handle the non-linear separability among different modulation classes.

- **Extreme Gradient Boosting (XGBoost) classifier:** The X-classifier has an ensemble learning model that constructs multiple decision trees sequentially to minimize classification errors. Unlike conventional boosting techniques, XGBoost incorporates regularization to improve generalization performance and reduce overfitting. The final prediction of the XGBoost classifier

is expressed as

$$\hat{y}_i = \sum_{k=1}^{n} f_k(\mathbf{x}_i), \quad f_k \in \mathcal{F} \tag{2.7}$$

where f_k represents the k^{th} decision tree and n denotes the total number of trees. The objective function minimized during training is given as

$$\mathcal{L} = \sum_{i} l(y_i, \hat{y}_i) + \sum_{k} \Omega(f_k) \tag{2.8}$$

where $l(\cdot)$ denotes the loss function and $\Omega(\cdot)$ represents the regularization term that controls the complexity of the model. The XGBoost classifier efficiently captures complex non-linear relationships among modulation features and improves classification performance under varying channel conditions.

Table 1. Design parameters for a dataset preparation phase: a typical indoor scenario.

Parameters	**Values**
Scenario	Indoor
Length and Width (meters)	10.9×9.2
# Transmitter positions	2
# Receiver locations	50
Distance between the transmitters (meters)	10
Spacing between the receivers (meters)	1
Received power range (dBm)	{-55,-75}

The different hybrid CNN models and CNN model are generated using the training dataset created for the indoor scenario. Then, these trained models (CNN, D-CNL-R, D-CNL-S, and D-CNL-X) are evaluated with the testing dataset collected from the 50 receiver positions of two transmitter locations for AMC.

3. Research methodology

The RFSC dataset for seven modulation classes have been prepared and transmitted using the GNU platform in the DR and received through SDR in the Wi-Guy. The dataset in [43] has been processed and generated using two transmitter locations and 50 receiver positions given in Table 1. Firstly, 50,000 I/Q samples have been captured for each modulation signal. To reduce computational complexity and memory overhead, the received signals have decimated using a factor of D=12. From the processed data, 6272 samples (1568×4) have been selected for each modulation class and converted into uint8-based image representations suitable for CNN-based feature extraction. The conversion gives efficient spatial feature learning while maintaining the essential modulation characteristics required for AMC. Although uint8 quantization and image transformation may introduce minor information loss, the representation achieves an effective balance between computational efficiency and classification performance while preserving sufficient discriminative features for reliable modulation classification. The D-CNL-M model has been generated and validated using image dataset. Then, the proposed

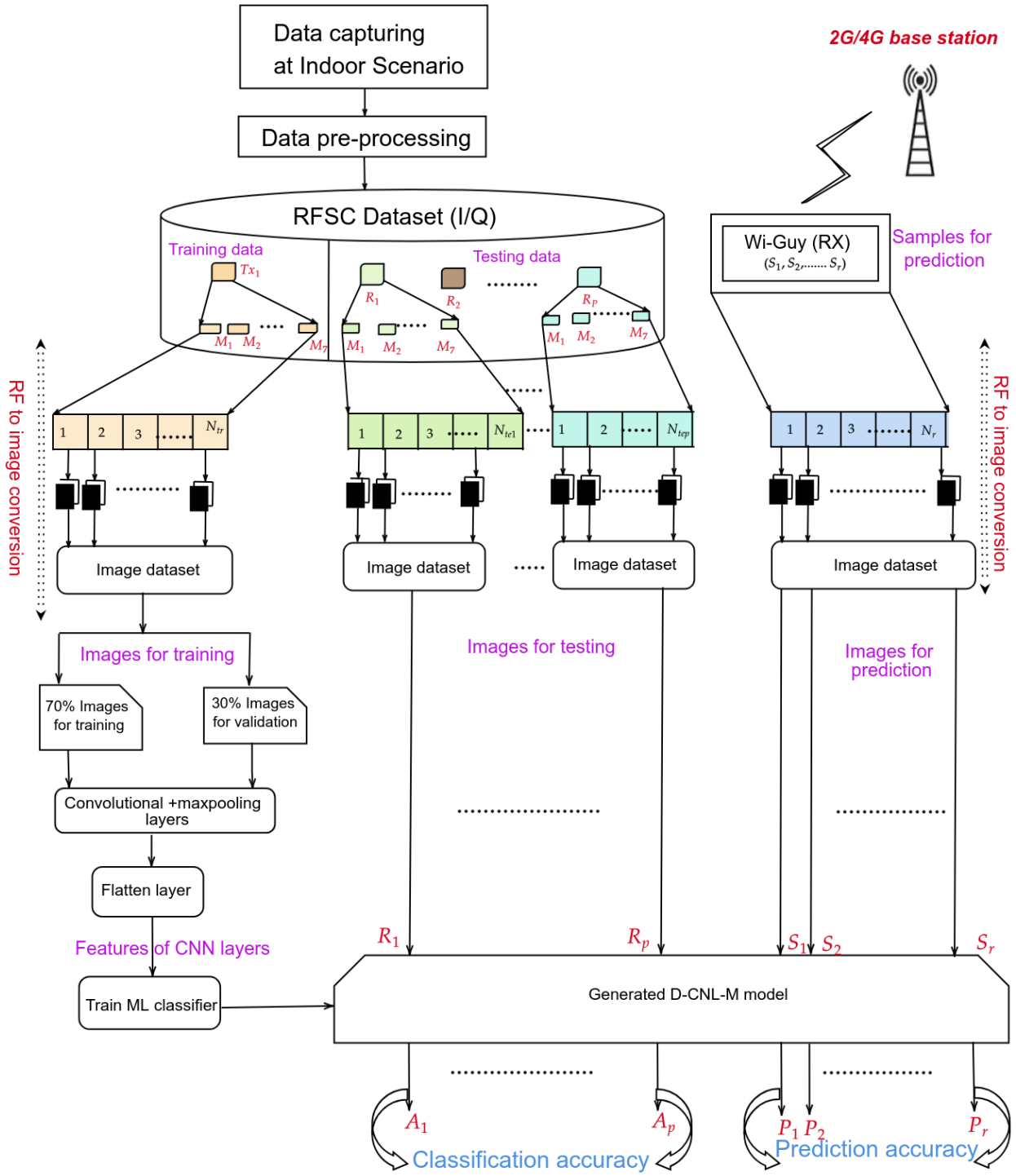


Figure 3. Proposed flow diagram for modulation classification/recognition using the D-CNL-M model.

model has been tested with dataset received from 50 different receiver positions. Moreover, the proposed model has been used to identify/recognize the type of modulation in real-time signals such as GMSK/QPSK, respectively. Interestingly, validation is performed within the training process for each convolutional layer during the "feature extraction" phase. The RF classifier works on the previous decision tree output in the "classification" stage. The proposed flow diagram for AMC is shown in Figure 3.

In our work, the D-CNL-M extracts higher representative features with lesser dimension using convolutional kernels and max pooling operations. However, each feature corresponds to the output of a trained convolutional layer. Interestingly, the number of convolutional layers (features) has been treated as a hyperparameter and introduces a higher degree of arbitrariness in the proposed work. However, we adopted an M-classifier to choose various features from several convolutional layers in the final classification performance. The detailed modulation classification procedure of the D-CNL-M model is described in Algorithm 1.

Table 2. Simulation parameters.

Parameter	Symbol	Value
Transmit signal power	p_t	10 dBm
Carrier frequency	f_c	810 MHz
Sampling frequency	f_s	2.4 MSps
Decimation value	D	12
Modulation classes	M	16QAM, 64QAM, BPSK, CPFSK, GFSK, GMSK, and QPSK
Number of signals per class	K	500
Samples length	N	6272
Training samples	X_{tr}	10,500 (75%)
Testing samples	X_{te}	3,500 (25%)
Batch size	B	128
Epochs number	-	70
Input image size	$(I_H \times I_W \times I_C)$	$28 \times 28 \times 2$
Signal to Noise Ratio	SNR	-5 to 10 dB
R-classifier trees	n	50
R-classifier random states	-	42
SVM kernel type	-	Radial Basis Function (RBF)
XGBoost evaluation metric	-	logloss

4. Simulation results and discussions: Modulation classification

We use the RFSC [44] dataset to evaluate the performance of the modulation task. In this work, we compare the performance of the CNN model with three hybrid CNN models. The system parameters are listed in Table 2. The training dataset (10,500) is split into two parts: 7,350 training images and

Algorithm 1 Modulation classification algorithm using D-CNL-M model

Input: Convolutional layers (l), receiver positions (p), and Transmitter locations (TX).

Conversion of I/Q ($r_b[n]$= $r_{b,I}[n]$ +j $r_{b,Q}[n]$) features into Image matrix

for $p \leftarrow 1$ **to** 50 **do**
  for $K \leftarrow 1$ **to** 7 **do**
    - Take v^{th} complex dataset vector :$r_{Kv} \epsilon\ C^N$
    - Transform complex into real valued vectors $x_{Kv} \epsilon\ 2N$
    - Translate x_{Kv} into matrix format, $R^{I_H \times I_w \times I_c}$
    - Convert x_{Kv} into images (I).

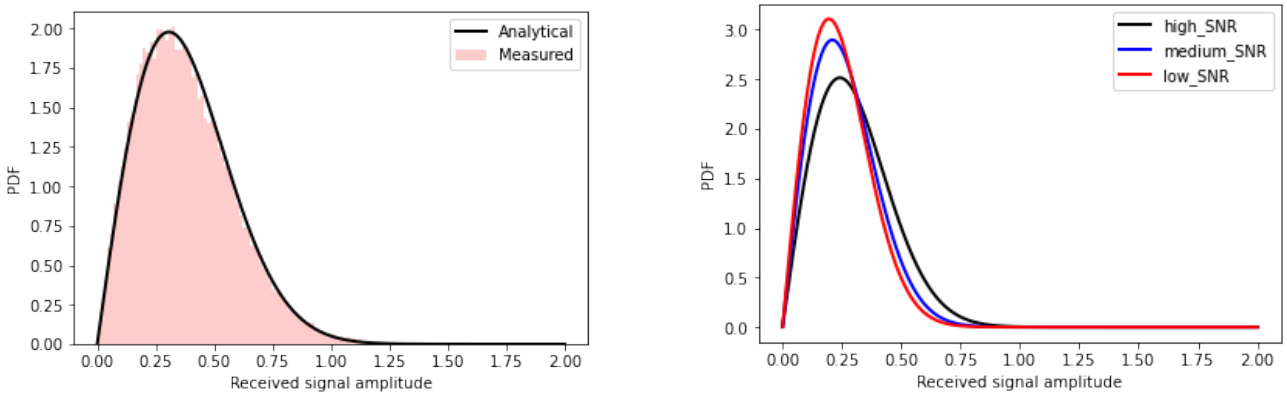
  end
end
while $TX = 1$ **do**
  input signal = Input (shape=(I_H, I_w, I_c))
  for $l \leftarrow 1$ **to** 3 **do**
    fe=Sequential()
    fe.add(Conv2D$_l$(filters, kernel_size, activation, padding))
    fe.add(Maxpooling2D$_l$(kernel_size, padding))
  end
  fe.add(Flatten())
  x =fe.output
  for $Epochs \leftarrow 1$ **to** 50 **do**
    update weights $\leftarrow$ $optimize$: $entropy$
    x_for_RF = fe.predict (x)
    RF_model = RandomForestClassifier()
    RF_model.fit(x_for_RF, y)
    SVM_model = SVMClassifier(kernel=RBF)
    SVM.fit(x_for_RF, y)
    XGB_model = XGBoostClassifier()
    XGB.fit(x_for_RF,y)
  end
  D-CNL-M model has been generated
end
for $p \leftarrow 1$ **to** 50 **do**
  Classification accuracy (A) for model D-CNL-M:

$$A(\%) = \frac{\sum_{i=1}^{M} (\hat{y}_i == y_i)}{\sum_{i=1}^{M} y_i} \times 100\%.$$

end

Output: 1. Testing classification accuracy (A)

3,150 validation images. For all CNN models, Adam optimizer has been adopted, and the categorical cross-entropy function is used to estimate the model loss during the training process. The Keras 1.18 library with the Tensorflow backend is used to train and test each CNN model. A six-core i5 processor personal computer with Ubuntu 18.04.2 is used for simulation purposes.



(a) Histogram plot of received signal at high SNR **(b)** Channel distribution for three SNR scenarios

Figure 4. Channel impairments: Multipath fading.

4.1. Fading channel

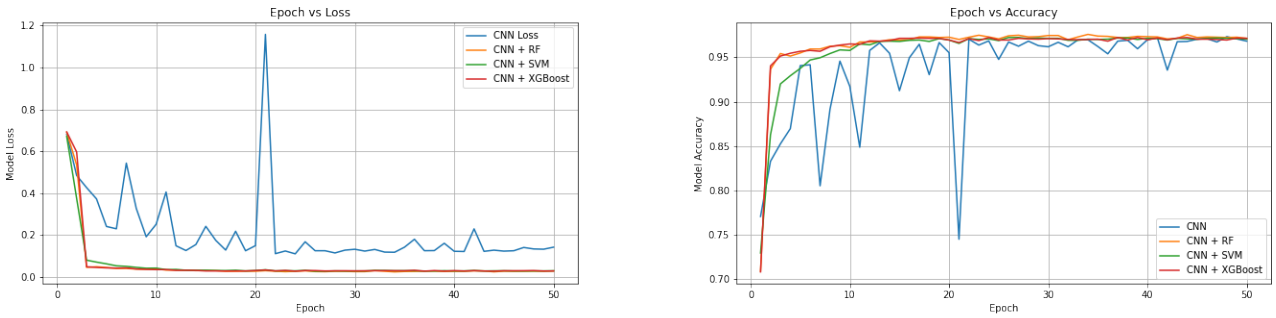
Our experiment is conducted under the indoor scenario and the received signal $r(t)$ with an amplitude 'α' following the Rayleigh distribution. The histogram plot of the received signal for the high SNR scenario is shown in Figure 4a. We find that the Probability Density Function (PDF) of captured samples follows the theoretical PDF. The PDF of a Rayleigh distribution is given in [45],

$$p(\alpha) = \frac{A}{\sigma^2} \exp\Big(- \frac{A^2}{2\sigma^2}\Big) \tag{4.1}$$

where, $A = \sqrt{I^2(t) + Q^2(t)}$ is the received signal amplitude α, $\sigma^2 = \mathbb{E}[\alpha^2]$ is the variance. The PDF of the received signal at different SNR scenarios is shown in Figure 4b. It can be seen that for Single Input Single Output (SISO) configuration at medium-low SNR scenarios, the channel response possesses a flat Rayleigh fading distribution and gradually moves closer to the Rician distribution for high SNR scenarios.

Indoor channels can be attenuated by penetration losses through walls and floors, resulting in multipath propagation. The received signal $r(t)$ components arrive in the form of clusters with a significant number of path delays. Rarely, the indoor channel typically behaves as a Rician channel as it receives LOS components alone. In the absence of LOS, Rayleigh fading becomes dominant in the adopted indoor scenario.

Figure 5a presents the nature of the model performance loss for CNN, and D-CNL-M models under various epoch values. It can be observed that the models does not experience any overshoot beyond the 20^{th} epoch. The results suggest that the models have been perfectly fit with hyper parameters as listed



(a) Model loss for CNN models

(b) Model accuracy for CNN models

Figure 5. Performance analysis of various CNN models.

in Table 2. Figure 5b presents the nature of the model accuracy for pre-defined, CNN, and D-CNL-M models under different epochs. After feature extraction, the similarity among modulation classes have become apparently recognised by D-CNL-R and it can achieve superior performance than the other models even for a lesser epoch value. Moreover, the model accuracy of all models still exceeds 90% with higher epoch values.

4.1.1. Training overhead (OH)

The offline training process has been performed to generate various hybrid models and baseline CNN using the training dataset for all seven different modulation classes, listed in Table 3. The training overhead can be calculated as:

$$OH = (No.of.parameters)\times (No.of.epoch) \times (No.of.samples) \tag{4.2}$$

Table 3. Offline training overhead for different CNN models.

Parameter	No.of Parameters	Generation time (minutes)	Training overhead	Training accuracy (%)	Inference time (minutes)
Model	**TX1&TX2**	**TX1&TX2**	**TX1&TX2**	**TX1&TX2**	**TX1&TX2**
CNN	5,91,751	25	$2.36 \times 10^{11}, (1\times)$	93	20
D-CNL-R	6,00,117	25	$2.21 \times 10^{11}, (1.02\times)$	99	14
D-CNL-S	5,92,344	27	$2.18 \times 10^{11}, (1.01\times)$	96.4	30
D-CNL-X	5,92,451	28	$2.19 \times 10^{11}, (1.01\times)$	97.3	28

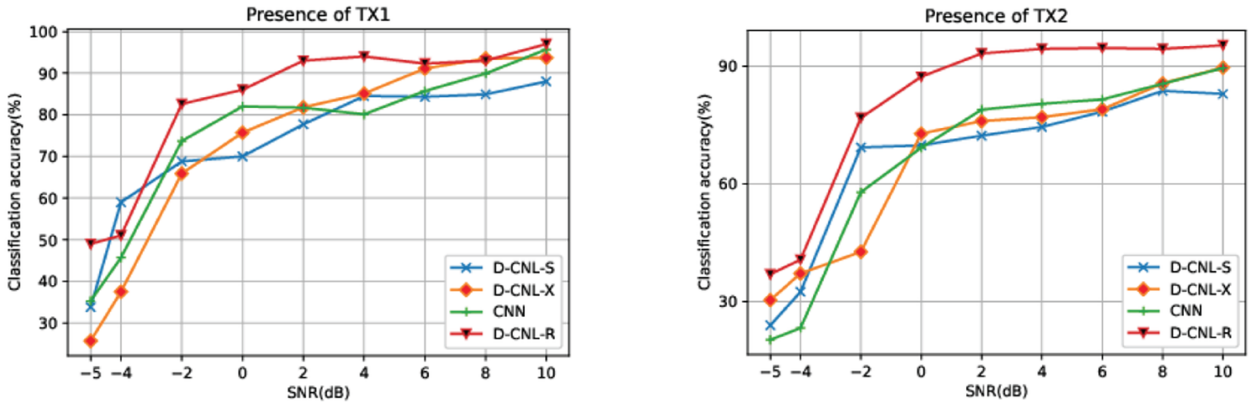
We can see that baseline CNN achieves the lowest computational overhead of 2.36×10^{11} with 5,91,751 trainable parameters. The hybrid models significantly increase the model complexity due to the integration of additional ML classifiers such as RF, SVM, and XGBoost. Moreover, it has been observed that the D-CNL-R model achieves training accuracy of up to 99% at the expense of less computational complexity owing to the ensemble structure of multiple decision trees in the RF classifier. Hence, the model generation time and training overhead of the D-CNL-R model appear to

be low compared to other models. The order of complexity for CNN model [46] is defined for L layer as: $O(W) = O(((\sum_{l=1}^{L}(n_{il}n_{fl}n_{kl}(n_{sl} - n_{kl} + 1)) + (n_s - n_k + 1)n_f) + (n_1 + n_1n_2 + n_2n_3))n_{ts})$, Here, n_{fl} and n_{kl} denote the amount of filters and filter size of the l^{th} layer. n_{il} is the number of input samples, n_f is a number of filters, n_{hl} is the number of hidden units, n_{ol} is the amount of output samples, n_{ts} is the number of time steps involved in the training process, L is the number of layers. n_1, n_2, n_3 represent the number of neurons in the first, second, and third layers of NN, respectively. For a D-CNL-R model (convolutional layers, flatten layer, and R-classifier), the order of complexity [47] is given by $O(W) = O(((\sum_{l=1}^{L}(n_{il}n_{fl}n_{kl}(n_{sl} - n_{kl} + 1))) + (n_s - n_k + 1)n_f + (kdn \log n))n_{ts})$, where n, d and k denote the number of data samples in the training dataset, amount of features, and a number of trees, respectively. Furthermore, the time required for the trained model to classify the modulation class during testing with all 50 different positions of the receiver, with the two different transmitter locations also included in the table.

4.2. Performance analysis of testing dataset

4.2.1. Classification accuracy

The proposed hybrid models have been compared with baseline CNN model under various SNR scenarios. Figures 6a and 6b show the overall testing accuracies for D-CNL-S, D-CNL-X, CNN, and D-CNL-R from -5 dB to 10 dB in the presence of TX1 and TX2. From Figure 6a, we observe that the testing accuracy for D-CNL-R is better than other models for various SNR conditions. At a SNR value of 10 dB, we have noticed that there is a similar performance between CNN and D-CNL-R models.



(a) TX1 **(b)** TX2

Figure 6. Comparison of classification accuracy for models in the TX1 and TX2 scenario.

We can also observe that the D-CNL-R model achieves 97% accuracy at 10 dB, compared with other models. Figure 6b shows the overall testing accuracies for various models from -5 dB to 10 dB in the presence of TX2 alone. It can be seen that the testing accuracy of all four models progressively increases and remains stable as SNR increases gradually. Among the four models, the D-CNL-R shows the highest testing accuracy with increasing SNR. For a SNR of 10 dB, D-CNL-R has a testing accuracy of 95% than the other models.

Table 4 summarizes the performance analysis of the hybrid CNN models in three SNR scenarios: high (SNR=10 and 6 dB), medium (SNR= 0 and 4 dB) and low (SNR= -5 and -1 dB). At low SNR (-5

dB), D-CNL-R achieves 49.5% and 15% accuracies in the presence of TX1 and TX2, respectively. At high SNR (10 dB), D-CNL-R provides 97.5% and 95.3% classification accuracies with TX1 and TX2, respectively. Moreover, D-CNL-S and D-CNL-X provide a similar accuracy value at a medium SNR value of 4 dB.

Table 4. Comparison of testing accuracies between the models for three cases of SNR.

Case	SNR (dB)	CNN (%)		D-CNL-S (%)		D-CNL-X (%)		D-CNL-R (%)	
		TX1	TX2	TX1	TX2	TX1	TX2	TX1	TX2
Low	-5	35.3	14.3	33.8	14.0	25.7	14.4	49.5	15.0
	-1	50.7	30.9	58.8	60.3	60.9	40.7	72.8	70.8
Medium	0	82.0	69.2	70.7	69.8	75.7	72.8	86.0	87.3
	4	80.1	80.4	84.5	76.5	85.0	77.6	94.0	94.4
High	6	85.7	81.5	84.3	78.4	91.1	79.5	93.0	94.6
	10	95.7	89.5	88.2	82.9	93.7	89.0	97.5	95.3

4.2.2. Confusion matrix metrics

For each K signal, samples in the test dataset for the m-th modulation class ($m = 1, 2, ..., M$), statistics such as true positive (TP_m), true negative (TN_m), false positive (FP_m) and false negative (FN_m), are computed as follows:

- If any one signal (K =500) is identified as a particular m-th class found to be matched with test data of the same m-th class, it is represented as,

$$TP_m = C_{mm}, \tag{4.3}$$

  where C_{mm} is a confusion matrix element that gives the total count of the correctly identified class label ($\hat{y}_i$).
- If a signal identified as a particular m-th class is unmatched with the test data of the same m-th class, it is represented as,

$$FP_m = \sum_{l=1}^{n} C_{lm} - TP_m, \tag{4.4}$$

  where n is the total number of classes and C_{lm} is the l-th row of the m-th modulation class confusion matrix element.
- If a signal is unidentified in a particular m-th class matched with test data of the same m-th class, it is denoted as

$$FN_m = \sum_{l=1}^{n} C_{ml} - TP_m, \tag{4.5}$$

  where C_{ml} is m-th modulation type with its corresponding l-row confusion matrix elements.

Table 5. Confusion matrix metrics evaluation between the models.

Parameter	Model Evaluation time (minutes)	Classification accuracy (%)		F1score (%)		Precision		Recall	
Model	**TX1&TX2**	**TX1**	**TX2**	**TX1**	**TX2**	**TX1**	**TX2**	**TX1**	**TX2**
CNN	40	61.6	67.7	58.4	64.5	60.1	66.2	61.3	67.1
D-CNL-R	34	65.3	68.4	62.0	62.5	62.8	67.8	64.5	69.2
D-CNL-S	36	63.7	69.3	60.6	63.3	62.7	68.2	59.2	70.4
D-CNL-X	37	65.5	70.3	65.6	63.3	65.7	69.4	62.2	68.4

These intermediate statistics are also used to compute precision (P_m), recall (R_m), and ($F1score$):

$$P_m = \frac{TP_m}{TP_m + FP_m}, \tag{4.6}$$

The statistics "Recall" used to derive the classification probability of the $m - th$ class:

$$R_m = \frac{TP_m}{TP_m + FN_m}. \tag{4.7}$$

Here, $P = \frac{\sum_{m=1}^{M} P_m}{M}$, $R = \frac{\sum_{m=1}^{M} R_m}{M}$, and $F1 = 2 \cdot \frac{P \cdot R}{P+R}$.

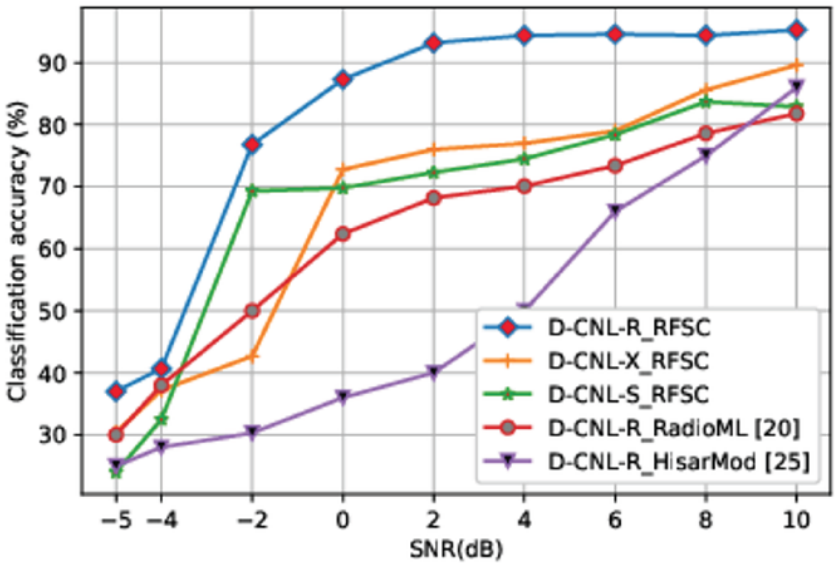


Figure 7. Comparison of classification accuracies across datasets.

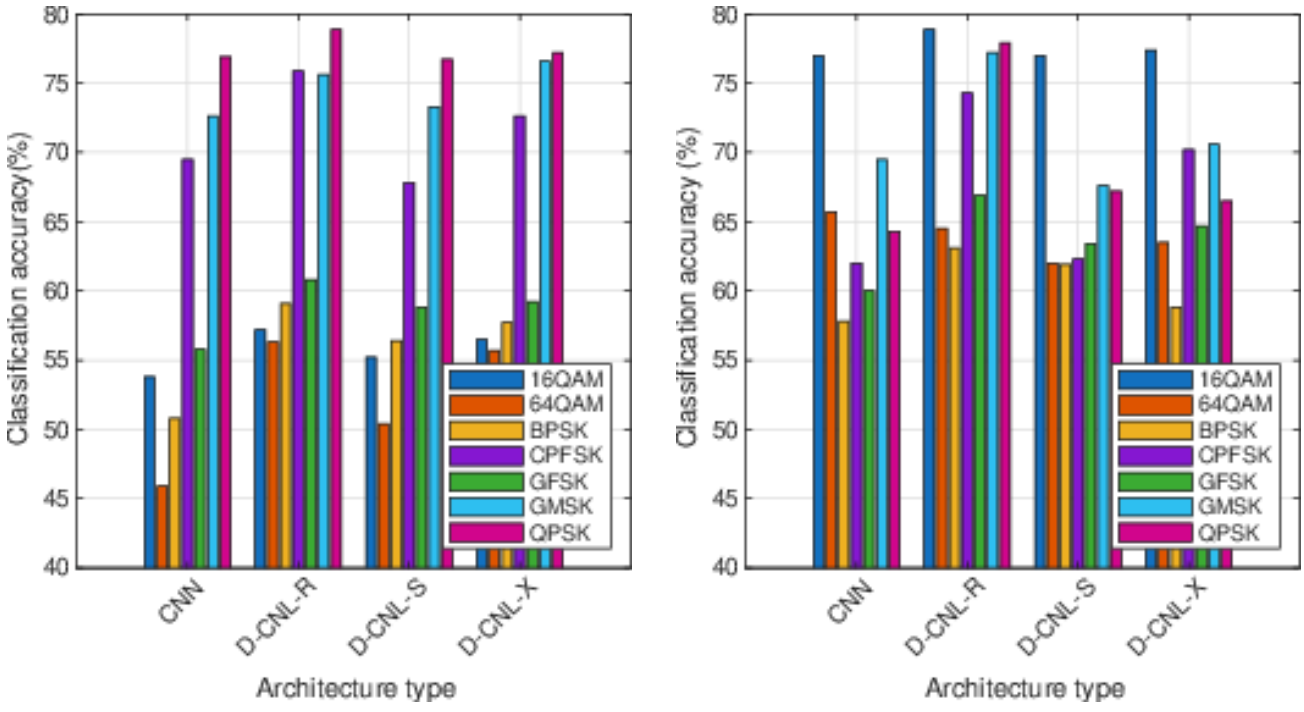
In order to carry out detailed model performance, different confusion metrics such as accuracy, precision, recall, and F1 score averaged over the testing dataset collected from 50 receiver positions of two transmitter locations are compared and listed in Table 5. It is revealed that the hybrid models yield the value of the metric as $\geq 62.0\%$. Moreover, we also notice that D-CNL-R and D-CNL-X models

achieve similar performance metrics. Furthermore, it is observed that the high Recalling capability of D-CNL-R produces better F1score, precision, and classification accuracy with lesser execution time compared to the baseline CNN and other hybrid models.

Figure 7 compares the classification performance of CNN models trained and tested on different datasets such as RFSC, RadioML, and HisarMod under varying SNR conditions ranging from −5 dB to 10 dB. The D-CNL-R model, developed using the RFSC dataset, consistently achieves higher classification accuracy compared to models trained on other datasets. This demonstrates the robustness and adaptability of the proposed model for different signal datasets and across environments.

4.3. Overall classification accuracy

Figure 8a illustrates the performance of seven modulation classes with all four hybrid models in terms of average classification accuracy (averaged over all receiver positions) in the presence of TX1. The D-CNL-R model performs well for the modulation classes 16QAM, CPFSK, and QPSK. The improvement is mainly due to the efficient decision boundary formation of the RF classifier, which effectively utilizes the deep features extracted from the CNN model. Furthermore, the D-CNL-S architecture also demonstrates improved classification performance over the baseline CNN model for various modulation types. However, its performance slightly degrades for higher-order modulation schemes compared to D-CNL-R and D-CNL-X due to the increased overlap among constellation features under noisy channel conditions.



(a) TX1 **(b)** TX2

Figure 8. Comparison of classification accuracy with CNN, D-CNL-R, D-CNL-S, and D-CNL-X models for TX1 and TX2.

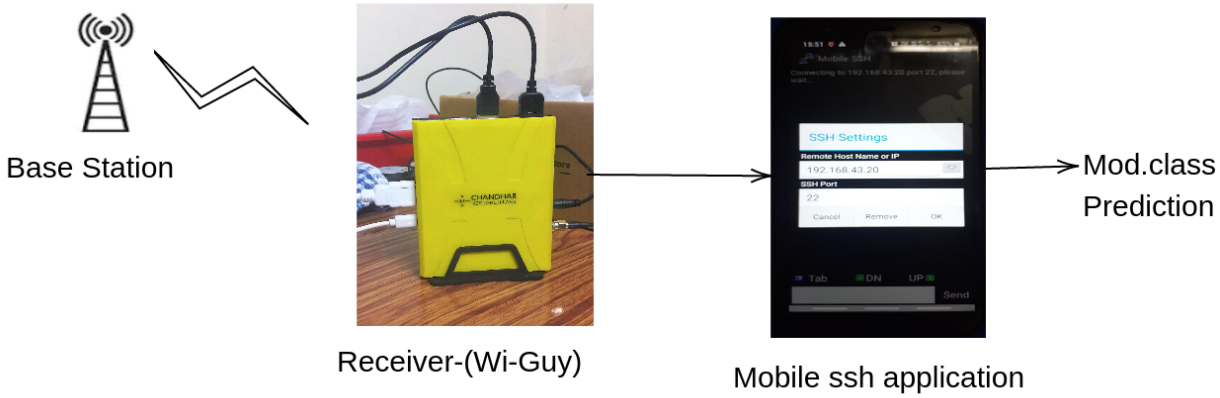


Figure 9. Measurement setup for real-time RF signal recognition.

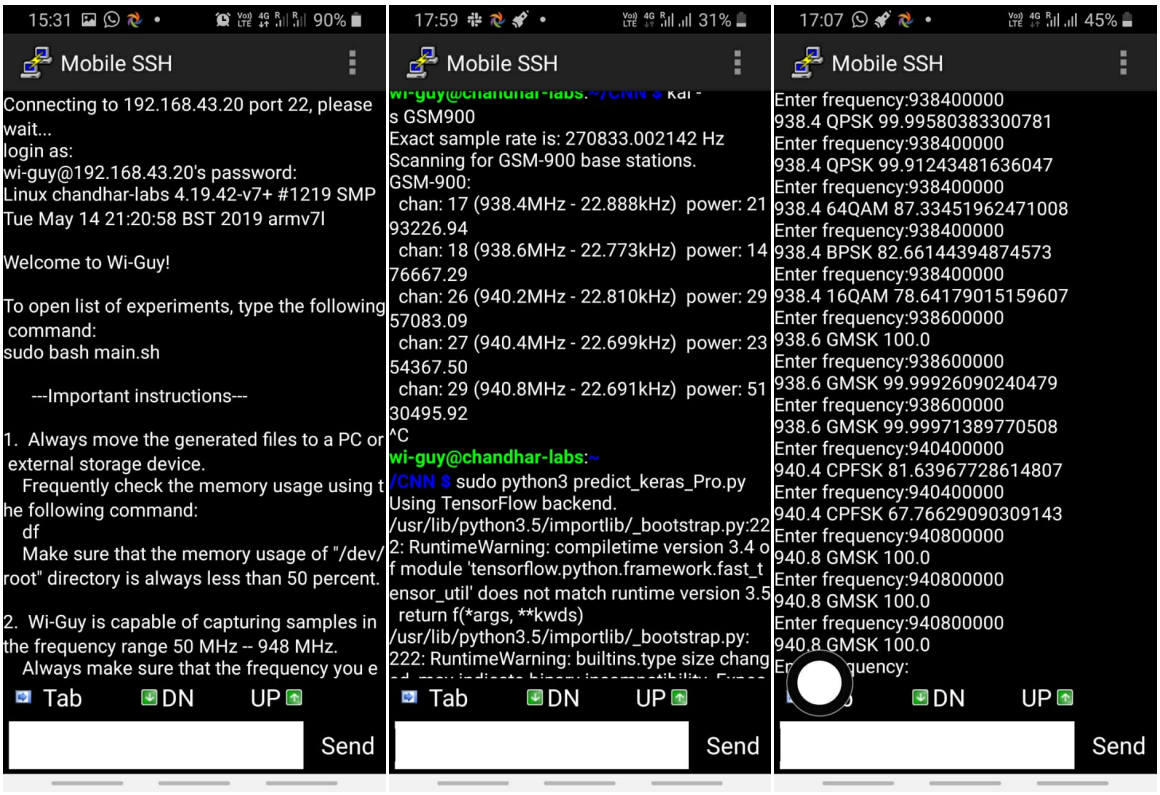
Figure 8b shows the average recognition accuracy performance using the TX2. The D-CNL-R and D-CNL-X models consistently achieve better performance. In particular, the D-CNL-R model achieves 78.9% accuracy for 16QAM and 77.9% for QPSK, whereas the D-CNL-X model achieves improved performance for CPFSK and GMSK modulation schemes. Hybrid architectures significantly improve modulation classification accuracy compared to the standalone CNN model. Among all, the D-CNL-R model achieves the best overall performance due to its efficient learning of complex non-linear features extracted by the CNN. The results demonstrate that combining deep feature extraction with traditional ML classifiers enhances robustness and generalization under varying transmitter positions and channel conditions.

5. Experimental results and discussions: Modulation recognition

The experimental setup to identify the modulation type in the real-time signal is shown in Figure 9. A valid tested model (CNN/D-CNL-R) is loaded into the Wi-Guy and accessed via smartphone using the Mobile SSH app. The real-time 2G-GSM and 4G-LTE signals from the nearby base stations are captured via the mobile ssh app by entering the IP address and port number of the Wi-Guy, as shown in Figure 10a. Now, we enter the following command in the Wi-Guy terminal to scan the active channel in the GSM 900 band.

In SDR, the kalibrate (kal) command displays the active GSM channel Absolute Radio Frequency Channel Number (ARFCN), Channel bandwidth, along with frequency offset variation, and its corresponding transmitter power values, as shown in Figure 10b. Now, the model adopts collected RF signals from the active GSM-900 channel to predict the modulation class in the RF transmission, as shown in Figure 10c. The modulation recognition performance of generated model is verified for a different location, as shown in Figure 11, and its corresponding recognition accuracies for the active frequencies scanned at indoor and outdoor scenarios are listed in Table 6.

We observe that the CNN model correctly predicts the modulation type as GMSK in 2G channels, whereas the D-CNL-R model fails to correctly predict the GMSK class. Furthermore, we extend this work to predict the modulation class for the 883 MHz (4G) channel. It has been noticed that the



(a) Credential screen **(b)** Scanning active bands **(c)** Modulation recognition

Figure 10. Modulation recognition of RF samples captured from nearby stations in Wi-Guy Receiver using the mobile ssh app.

Table 6. Recognition performance of real-time 2G/4G signals captured from the nearest base stations.

	Indoor		Outdoor			
Locations	L1		L2		L3	
Model Frequency (MHz)	CNN	D-CNL-R	CNN	D-CNL-R	CNN	D-CNL-R
940.8 (2G)	GMSK (100%)	BPSK (40%)	GMSK (91%)	QPSK (40%)	GMSK (94%)	CPFSK (90%)
956.4 (2G)	GMSK (89%)	GMSK (82%)	GMSK (57%)	QPSK (48%)	GMSK (90%)	GMSK (50%)
2150(4G)	QPSK 98%	QPSK 100%	QPSK 95%	QPSK 100%	BPSK 97%	BPSK 100%
883 (4G)	BPSK 97%	BPSK 100%	QPSK 97%	QPSK 100%	QPSK 96%	QPSK 100%

Figure 11. Modulation recognition location(L1: Lab, L2: Main Road, and L3: Park, Shenoy Nagar, Chennai, Tamil Nadu, India [48].

CNN and D-CNL-R models predict the QPSK class with recognition accuracies of ≥ 96% and 100%, respectively. As is shown, the Physical Broadcast Control Channel (PBCH), Cell Reference Signals (CRS), Secondary Synchronization Signals (SSS), and Primary Synchronization Signals (PSS) in 4G uses QPSK modulation for the control signal transmission. The data are transmitted either by QPSK or any one type of QAM: 16QAM, 64QAM modulation as specified in the 3GPP standard [49]. The work can be further extended to the 5G cellular system by including the 256 QAM modulation class.

6. Conclusions

This paper presented a novel D-CNL-M classifier model for AMC. In this approach, the D-CNL-M utilizes the RFSC dataset to extract more informative features, which are then passed via ML classifiers such as RF, SVM, and XGBoost. Initially, RF I/Q signal samples have been transformed into image representations, and deep spatial features have been extracted using a CNN architecture consisting of convolutional and max-pooling layers. The extracted features were subsequently classified using RF, SVM, and XGBoost classifiers to improve modulation recognition performance. Moreover, we also found that signal propagation follows a proper Rayleigh channel distribution in the indoor scenario. The phase-related modulation classes, such as 16QAM, 64QAM, and QPSK, have been well classified with the D-CNL-R model. Finally, the efficiency and complexity of the D-CNL-R model have been demonstrated and verified using the RFSC dataset. Finally, we presented a recognition performance for real-time signals and verified it with the proposed D-CNL-R model. The low complexity and highly efficient feature learning capabilities of the D-CNL-R model make it attractive for realizing intelligent

receivers in real-world scenarios such as IoT and Vehicle-to-Everything (V2X) applications for next-generation cellular systems. Moreover, benchmarking against transformer-based and attention-driven architectures is identified as a key direction for future work.

Author contributions

Tamizhelakkiya K: Conceptualization, Methodology, Software, Formal analysis, Investigation, Visualization, Writing-original draft preparation. C.T. Manimegalai: Supervision, Validation, Resources, Writing-review and editing.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

All authors declare that they have no conflict of interest.

References

1. Fu S, Yang F, Xiao Y (2020) Ai inspired intelligent resource management in future wireless network. *IEEE Access* 8: 22425–22433. https://doi.org/10.1109/ACCESS.2020.2968554
2. Zhang Z, Xiao Y, Ma Z, Xiao M, Ding Z, Lei X, et al. (2019) 6g wireless networks: Vision, requirements, architecture, and key technologies. *IEEE Vehicular Technology Magazine* 14: 28–41. https://doi.org/10.1109/MVT.2019.2921208
3. Tang F, Kawamoto Y, Kato N, Liu J (2020) Future intelligent and secure vehicular network toward 6g: Machine-learning approaches. *Proceedings of the IEEE* 108: 292–307. https://doi.org/10.1109/JPROC.2019.2954595
4. Salahdine F, Han T, Zhang N (2023) 5g, 6g, and beyond: Recent advances and future challenges. *Annals of Telecommunications* 78: 525–549.
5. Dobre OA, Abdi A, Bar-Ness Y, Su W (2007) Survey of automatic modulation classification techniques: classical approaches and new trends. *IET communications* 1: 137–156. https://doi.org/10.1049/iet-com:20050176
6. Zhu Z, Nandi AK (2015) *Automatic modulation classification: principles, algorithms and applications*, John Wiley & Sons.
7. Kulin M, Kazaz T, Moerman I, De Poorter E (2017) End-to-end learning from spectrum data a deep learning approach for wireless signal identification in spectrum monitoring applications. *IEEE Access* 6: 18484 – 18501. https://doi.org/10.1109/ACCESS.2018.2818794
8. Zhang W, Zheng L, Xu Y, Wang G, Wu Y (2018) Supervised learning method for link adaptation algorithm in coded mimo-ofdm systems. In *IEEE 4th International Conference on Computer and Communications (ICCC)*, 414–419. https://doi.org/10.1109/CompComm.2018.8780721

9. Song M, Xin C, Zhao Y, Cheng X (2012) Dynamic spectrum access: from cognitive radio to network radio. *IEEE Wireless Communications* 19: 23–29. https://doi.org/10.1109/MWC.2012.6155873
10. Trinh HD, Giupponi L, Dini P (2018) Mobile traffic prediction from raw data using lstm networks. In *IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications(PIMRC)*, 1827–1832. https://doi.org/10.1109/PIMRC.2018.8581000
11. Tang B, Tu Y, Zhang Z, Lin Y (2018) Digital signal modulation classification with data augmentation using generative adversarial nets in cognitive radio networks. *IEEE Access* 6: 15713–15722. https://doi.org/10.1109/ACCESS.2018.2815741
12. Ujan S, Navidi N, Landry RJ (2020) Hierarchical classification method for radio frequency interference recognition and characterization in satcom. *Applied Sciences* 10: 4608. https://doi.org/10.3390/app10134608
13. Jagannath J, O'Connor D, Polosky N, Sheaffer B, Foulke S, Theagarajan LN, et al. (2017) Design and evaluation of hierarchical hybrid automatic modulation classifier using software defined radios. In *IEEE Annual Computing and Communication Workshop and Conference (CCWC)*, 1–7. https://doi.org/10.1109/CCWC.2017.7868362
14. Markovic GB, Dukic ML (2015) Joint cumulant estimate correction and decision for cooperative modulation classification by using multiple sensors. *Annals of telecommunications-annales des télécommunications* 70: 197–206. https://doi.org/10.1007/s12243-014-0437-4
15. Khan S, Islam N, Jan Z, Ud Din I, Rodrigues JJPC (2019) A novel deep learning based framework for the detection and classification of breast cancer using transfer learning. *Pattern Recognition Letters* 125: 1–6. https://doi.org/10.1016/j.patrec.2019.03.022
16. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, 1097–1105.
17. Lyu Z, Wang Y, Li W, Guo L, Yang J, Sun J, et al. (2020) Robust automatic modulation classification based on convolutional and recurrent fusion network. *Physical Communication* 43: 101213. https://doi.org/10.1016/j.phycom.2020.101213
18. Hagos MT, Kant S (2019) Transfer learning based detection of diabetic retinopathy from small dataset. *ArXiv preprint arXiv:1905.07203.*
19. Shi Q, Gong Y, Guan YL (2008) Asynchronous classification of high-order QAMs. In *IEEE Wireless Communications and Networking Conference*, 1188–1193. https://doi.org/10.1109/WCNC.2008.214
20. O'Shea T, West N (2016) Radio machine learning dataset generation with GNU radio. *Proceedings of the GNU Radio Conference* 1.
21. Guo J, He T, Lausen L, Li M, Lin H, Shi X, et al. (2020) Gluoncv and gluonnlp: Deep learning in computer vision and natural language processing. *Journal of Machine Learning Research* 21: 1–7.
22. Kim K, Akbar IA, Bae KK, Um J, Spooner CM, Reed JH (2007) Cyclostationary approaches to signal detection and classification in cognitive radio. In *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, 212–215. https://doi.org/10.1109/DYSPAN.2007.35

23. Fehske A, Gaeddert J, Reed JH (2005) A new approach to signal classification using spectral correlation and neural networks. In *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, 144–150. https://doi.org/10.1109/DYSPAN.2005.1542629

24. Chen M, Challita U, Saad W, Yin C, Debbah M (2019) Artificial neural networks-based machine learning for wireless networks: A tutorial. *IEEE Communications Surveys Tutorials* 21: 3039–3071.

25. Tekbıyık K, Ekti AR, Görçin A, Kurt GK, Keçeci C (2020) Robust and fast automatic modulation classification with CNN under multipath fading channels. *IEEE Vehicular Technology Conference (VTC)*, 1–6. https://doi.org/10.1109/VTC2020-Spring48590.2020.9128408

26. Peng S, Jiang H, Wang H, Alwageed H, Yao YD (2017) Modulation classification using convolutional neural network based deep learning model. In *26th Wireless and Optical Communication Conference (WOCC)*, 1–5. https://doi.org/10.1109/WOCC.2017.7929000

27. Peng S, Jiang H, Wang H, Alwageed H, Zhou Y, Sebdani M, et al. (2018) Modulation classification based on signal constellation diagrams and deep learning. *IEEE Transactions on Neural Networks and Learning Systems* 30: 718–727.

28. Alhazmi MH, Alymani M, Alhazmi H, Almarhabi A, Samarkandi A, Yao Y (2020) 5G signal identification using deep learning. In *Wireless and Optical Communications Conference (WOCC)*, 1–5. https://doi.org/10.1109/WOCC48579.2020.9114912

29. Shi J, Hong S, Cai C, Wang Y, Huang H, Gui G (2020) Deep learning-based automatic modulation recognition method in the presence of phase offset. *IEEE Access* 8: 42841–42847.

30. Teng CF, Chou CY, Chen CH, Wu AY (2020) Accumulated polar feature-based deep learning for efficient and lightweight automatic modulation classification with channel compensation mechanism. *IEEE Transactions on Vehicular Technology* 69: 15472–15485. https://doi.org/10.1109/TVT.2020.3041843

31. Roy D, Mukherjee T, Chatterjee M, Pasiliao E (2019) RF transmitter fingerprinting exploiting spatio-temporal properties in raw signal data. In *IEEE International Conference On Machine Learning And Applications (ICMLA)*, 89–96. https://doi.org/10.1109/ICMLA.2019.00023

32. Zhang D, Ding W, Zhang B, Xie C, Li H, Liu C, et al. (2018) Automatic modulation classification based on deeplearning for unmanned aerial vehicles. *Sensors* 18: 924. https://doi.org/10.3390/s18030924

33. Hong D, Zhang Z, Xu X (2017) Automatic modulation classification using recurrent neural networks. In *3rd IEEE International Conference on Computer and Communications (ICCC)*, 695–700. https://doi.org/10.1109/CompComm.2017.8322633

34. Tamizhelakkiya, Chandhar P, Gauni S (2021) Comparison of deep architectures for indoor rf signal classification. In *2021 International Conference on Emerging Techniques in Computational Intelligence (ICETCI)*, 107–112. https://doi.org/10.1109/ICETCI51973.2021.9574083

35. Fu X, Gui G, Wang Y, Gacanin H, Adachi F (2022) Automatic modulation classification based on decentralized learning and ensemble learning. *IEEE Transactions on Vehicular Technology* 71: 7942–7946. https://doi.org/10.1109/TVT.2022.3164935

36. Zhang X, Sun J, Zhang X (2020) Automatic modulation classification based on novel feature extraction algorithms. *IEEE Access* **8**: 16362–16371. https://doi.org/10.1109/ACCESS.2020.2966019
37. HackRF, `https://greatscottgadgets.com/hackrf/`.
38. Rtl-sdr, `https://www.rtl-sdr.com/`.
39. Chandhar, `https://www.chandhar-labs.com/products/wi-guy/`.
40. Wu P, Sun B, Su S, Wei J, Zhao J, Wen X (2020) Automatic modulation classification based on deep learning for software-defined radio. *Mathematical Problems in Engineering* 2020: 2678310. https://doi.org/10.1155/2020/2678310
41. Tamizhelakkiya K, Gauni S, Chandhar P (2023) Transfer learning based location-aided modulation classification in indoor environments for cognitive radio applications. *Radioengineering* 32. https://doi.org/10.13164/re.2023.0531
42. Eluri VK, Mazzuchi TA, Sarkani S (2021) Predicting long-time contributors for github projects using machine learning. *Information and Software Technology* 138: 106616. https://doi.org/10.1016/j.infsof.2021.106616
43. RFSC Dataset for modulation classification, `https://www.kaggle.com/datasets/chandharlabs/rfsc-dataset`.
44. Tamizhelakkiya, Chandhar P, Gauni S (2021) Comparison of deep architectures for Indoor RF signal classification. In *International Conference on Emerging Techniques in Computational Intelligence (ICETCI)*, 107–112. https://doi.org/10.1109/ICETCI51973.2021.9574083
45. Yang J, Mostafapour E, Aminfar A, Wang J, Huang H, Akhbari A, et al. (2019) Channel fading effect analysis on diffusion cooperation strategies over adaptive networks. *KSII Transactions on Internet and Information Systems* 13: 172–185. https://doi.org/10.3837/tiis.2019.01.010
46. Freire PJ, Osadchuk Y, Spinnler B, Napoli A, Schairer W, Costa N, et al. (2021) Performance versus complexity study of neural network equalizers in coherent optical systems. *Journal of Lightwave Technology* 39: 6085–6096. https://doi.org/10.1109/JLT.2021.3096286
47. Hassine K, Erbad A, Hamila R (2019) Important complexity reduction of random forest in multi-classification problem. In *International Wireless Communications & Mobile Computing Conference (IWCMC)*, 226–231. https://doi.org/10.1109/IWCMC.2019.8766544
48. Chandhar P, Tamizhelakkiya, Babu S, Krishnakumar V, Sivakumar A, Viswanathan W (2022) 4g spectrum occupancy analysis in chennai-india. In *International Conference on Emerging Techniques in Computational Intelligence (ICETCI)*, 132–135. https://doi.org/10.1109/ICETCI55171.2022.9921373
49. *3GPP TS 38.211 version 15.2.0 Release 15 for 5G NR*, Technical report, 3GPP, 2018.