Journal of Azzaytuna University (JAZU)
JAZU Homepage: https://azzujournal.azu.edu.ly
ISSN (Print):2523-1006, ISSN (Online): 3105-4420 DOI: 10.35778/x08ctx57

Performance Evaluation of Deep Learning Approaches for Automatic Modulation Classification: A Comparison between CNN and CLDNN Networks

Fathia A. Husain[1*], Abdul Latif Al-Shami[2], Mahmoud Oweili[3], and Tarek S. Abdou[4]
[1,2,3,4] Dept. of Electrical and Electronic Engineering Higher Institute of Engineering Technologies - Tripoli, Libya
fatehiahusain@gmail.com[1*], soulnight480@gmail.com[2], fog20man@gmail.com[3],atgps2015@gmail.com[4]

Corresponding Author:(*)

Article history: Received September 07th , 2026, Revised September 25th , 2026, Accepted September 30th , 2026

ABSTRACT: Automatic Modulation Classification (AMC) is considered as an important stage in wireless communication systems because it gives the receiver the ability to identify the modulation type of an unknown received signal. This paper proposes an AMC implementation that is based on two models of deep learning: 1-Dimensional Convolutional Neural Networks (CNNs) and Convolutional Long Short-Term Memory Deep Neural Networks (CLDNNs). It applies the Radio ML RML2016.10a dataset which contains 11 analog and digital modulations, 128 sample I/Q frames, and SNR levels ranging from -20 dB to +18 dB. There are two filter settings used in this experiment. The first one involves 50 CNN filters and 64 CLDNN convolutional/LSTM units while the second setting is a compact one and consists of 40 CNN filters and 54 CLDNN convolutional/LSTM units. Models were trained via Adam optimizer and categorical cross-entropy loss, learning rate 0.0007, batch size 128, 50 epochs, and 15% of validation set. This implementation performs accuracy evaluation for every SNR level and provides normalized confusion matrices.

Keywords: **Automatic modulation classification (ACM), CNN, CLDNN, LSTM, I/Q signals.**

الملخص:تتناول هذه الدراسة موضوع التصنيف التلقائي لأنماط التعديل (AMC) في أنظمة الاتصالات اللاسلكية، وهي عملية مهمة، حيث تتيح لأجهزة الاستقبال التعرف على نوع تعديل الإشارة المجهولة المستلمة .يواجه التصنيف التقليدي صعوبات بالغة نتيجة الضوضاء والتشويه في القنوات اللاسلكية، لا سيما عند انخفاض نسبة الإشارة إلى الضوضاء (SNR) وتداخل الأنماط ذات الخصائص المتشابهة .ولحل هذه المشكلات، يقدم البحث مقارنة ممنهجة بين نموذجين للتعلم العميق: الشبكات العصبية الالتفافية أحادية الأبعاد (1D–CNN) والشبكات الهجينة (CLDNN) التي تدمج الالتفاف مع خلايا الذاكرة الطويلة قصيرة المدى (LSTM) لتتبع التغيرات الزمنية للإشارات.تم تقييم النماذج باستخدام قاعدة بيانات RadioML RML2016.10aالتي تحتوي على 11 نمط تعديل تناظري ورقمي .كما تضمن البحث دراسة تأثير تقليص عدد المرشحات والوحدات بنسبة تتراوح بين 20% و25% بهدف خفض التعقيد الحسابي للمعلمات .أظهرت النتائج تفوق نموذج CLDNN الهجين بتحقيقه دقة قصوى تتراوح بين 88% و89.5% مقارنة بدقة بلغت 82% لنموذجCNN .كما أثبتت الدراسة أن تقليص معلمات النماذج يقلل التعقيد الحسابي بشكل كبير مقابل تراجع طفيف للغاية في الدقة لا يتجاوز 1% إلى 1.5%، مما يوفر موازنة مثالية للتطبيقات اللاسلكية ذات الموارد المحدودة.

***الكلمات المفتاحية*: التصنيف التلقائي لأنماط التعديل(AMC)، CNN، CLDNN، LSTM، إشارات I/Q.**

I. INTRODUCTION

Automatic Modulation Classification (AMC) is used to identify the modulation type of a received radio signal without prior knowledge of the transmitter. It is an important task in software defined

 This is an open access article under the CC BY-SA license.

radio (SDR), cognitive radio, spectrum monitoring, signal intelligence, interference detection and adaptive wireless receivers [1][2].

However, the received signal suffers from noise and impairments in the channel. These Impairments may modify the amplitude, phase, frequency, or time of the signal. Examples include additive noise, fading, multipath propagation, frequency offset, phase offset, and sample-rate offset [3]. The problem is more difficult at low signal-to-noise ratio (SNR), where the noise power is close to or greater than the signal power. Short signal sequences also make it difficult to separate modulation types with similar characteristics. For example, QPSK and 8PSK can be confused when phase information is weak. QAM16 and QAM64 can also be confused when the amplitude levels are not clear [4][5].

Traditional AMC techniques usually follow either of the two approaches. The first one is Likelihood-based classification, wherein the received signal is matched with the statistical model associated with each modulation scheme. Another approach involves manually selected features, that may include amplitude, phase, frequency, higher-order statistics, cyclostationary characteristics, or wavelet characteristics. These techniques may perform very effectively, but they may require expert knowledge and may be sensitive to changes in the channel [7].

Deep learning offers another approach. Rather than choosing signal features by hand, we can train a neural network to learn features directly from raw I/Q signals. Convolutional neural networks are particularly well-suited to learning local features of waveforms. Recurrent networks like LSTM networks can learn temporal features of the signal. There is extensive literature on CNN-based modulation recognition for radio signals [6].

This paper is based on the provided Python implementation. It compares two models:

A 1D-CNN model that learns local temporal features from I/Q sequences. A CLDNN model that combines convolutional feature extraction with LSTM-based temporal learning.

The paper also compares two model-width settings:

- Standard configuration: 50 CNN filters and 64 CLDNN filters/LSTM units.
- Reduced configuration: 40 CNN filters and 54 CLDNN filters/LSTM units.
- The main goal is to study whether reducing the number of filters and LSTM units can lower the model complexity while keeping acceptable modulation-classification performance.

II. RELATED WORK

The development of Automatic Modulation Classification (AMC) has transitioned from rigidstatistical algorithms to adaptive, data-driven deep learning models. In accordance with standard literature classifications [1], we classify previous research into two major paradigms.

A) Classical Decision-Theoretic and Feature-Based AMC

Traditionally, AMC algorithms have been divided into decision-theoretic (likelihood-based) and feature-based (FB) pattern-recognition methods [10].

Likelihood-based (LB) techniques treat classification as a composite hypothesis testing problem, using probability density functions to evaluate received waveforms [11]. Standard decision metrics

include the Average Likelihood Ratio Test (ALRT), Generalized Likelihood Ratio Test (GLRT), and Hybrid Likelihood Ratio Test (HLRT) [15]. While LB methods are theoretically optimal (minimizing the probability of classification error), they require precise estimates of channel parameters (e.g., carrier phase offset, frequency offset, noise variance, and symbol timing). If any estimate has even a slight error, the performance degrades drastically. Furthermore, calculating multi-dimensional integrals over unknown parameters in real-time is computationally intensive, limiting their utility in real-time applications [16].

Feature-based (FB) approaches have been proposed to simplify the classification problem by extracting suboptimal but simpler handcrafted signal features, such as instantaneous descriptors (amplitude, phase and frequency), higher order statistics (HOS), cyclostationary features and wavelet transforms, some of others[12]. These features are then fed into standard machine learning classifiers like Support Vector Machines (SVM), K-Nearest Neighbours (KNN) or decision tree models [8]. However, handcrafted feature extraction is time-consuming and heavily depends on the prior expert knowledge in the domain. Moreover, FB approaches are not robust to dynamic wireless environments, where environmental degradation such as multipath fading, clock frequency offsets and low signal-to-noise ratios (SNR) degrades the discriminability of handcrafted descriptors [14].

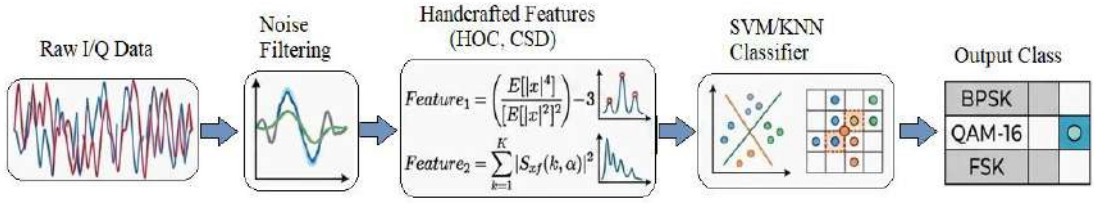


Fig. 1 Traditional Feature-Based ACM Pipeline.

B) Deep Learning-Based AMC

Deep learning architectures have overcome the limitations of manual feature engineering by automatically extracting multi-dimensional representations directly from raw baseband sequences [9].

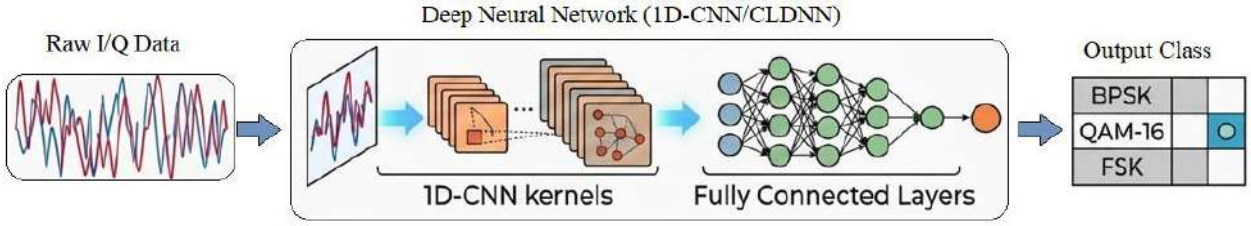


Fig. 2 Deep Learning End-to-End ACM Pipeline.

The following studies are closely related to the present work and provide the main background for the proposed CNN and CLDNN comparison. [6] proposed a CNN-LSTM dual-stream architecture for automatic modulation classification that learns local signal features and temporal relationships. The CNN extracts spatial features, while LSTM learns feature changes over time. This model,

which combines various signal representations for classification, is simpler compared to the CLDNN model used in this work, as it incorporates only one Conv1D layer, max pooling, two LSTM layers, dropout, flattening, and an 11-class softmax output. Additionally, this study examines the effects of reducing CLDNN width from 64 to 54 filters and LSTM units on performance and complexity. [8] developed a CNN architecture for modulation classification in cognitive radio networks, leveraging raw I/Q sequences. Their method allows CNNs to learn features like local amplitude changes and phase transitions without manual feature extraction. The current study parallels this by using direct I/Q signal processing with a 128×2 input matrix for the in-phase and quadrature components. It includes several CNN layers—zero padding, three convolutional layers, max-pooling, dropout, a dense layer with 70 units, and an 11-class softmax layer while also contrasting performance with a CLDNN model and examining CNN widths of 50 and 40 filters.

developed a lightweight temporal hybrid neural network aimed at robust automatic modulation classification, emphasizing performance while reducing computational costs. This work is significant for real-world applications like SDR platforms, edge devices, and embedded cognitive radio nodes, where processing power and memory constraints are critical. Their goal is to maintain accuracy without compromising model efficiency [5]. This work focuses on a filter-reduction strategy by decreasing the CNN width from 50 to 40 filters and the CLDNN width from 64 to 54, including convolutional filters and LSTM units. This simplification allows for a direct comparison between higher-capacity and reduced-complexity models, aiming to assess whether smaller architectures can maintain acceptable SNR-dependent accuracy with fewer parameters and memory usage.

In the research of [2] a CNN-LSTM hybrid architecture study has advanced automatic modulation classification (AMC) evaluation beyond simulated datasets, incorporating practical wireless conditions through software-defined radio experiments. This approach accounts for factors like channel fading, hardware effects, frequency mismatch, noise, and synchronization issues.

This work is relevant because the present study focuses on Software Defined Radio (SDR) and cognitive-radio applications, evaluating CNN and CLDNN models across a signal-to-noise ratio (SNR) range of −20 dB to +18 dB. The system can utilize the RML2016.10a dataset or generate synthetic I/Q signals if the dataset is unavailable. Although it lacks real over-the-air experiments, the analysis includes SNR evaluations and confusion matrices for simulation-based assessment under noisy conditions. Future directions involve extending these implementations to facilitate SDR-based over-the-air signal collection and testing. The reviewed studies show that CNN, CNN-LSTM, and CLDNN-related methods are effective for AMC. However, recent work often uses advanced structures such as dual streams, bidirectional recurrence, hybrid temporal layers, and practical SDR testbeds. The present study focuses on a simpler and reproducible comparison between CNN and CLDNN models. The main difference of the present work is the controlled filter-width analysis:

CNN: 50→40 filters

CLDNN: 64→54 convolutional filters and LSTM units.

Convolutional filters and LSTM units. The models are tested using the same I/Q input format, optimizer, training settings, SNR range, and evaluation procedure. This makes it possible to study the effect of reducing model width on classification accuracy, low-SNR robustness, confusion patterns, and computational complexity.

III. METHODOLOGY

A) Dataset and Signal Representation

The dataset used in this work is RadioML RML2016.10a[12]. It contains 11 modulation formats, including eight digital modulations and three analog modulations. The dataset uses 20 SNR values from −20 dB to +18 dB with a 2 dB step. Each signal example is represented by 128 I/Q samples.

The modulation classes used in the implementation are: (8PSK,AM-DSB,AM-SSB,BPSK,CPFSK,GFSK,PAM4,QAM16,QAM64,QPSK,WBFM).

The received baseband signal can be written as:

$$s(t) = s_I[n] + js_Q[n], \quad n=0,1,.....127 \tag{1}$$

where $s_I[n]$is the in-phasecomponent, $s_Q[n]$is the quadrature component of the received complex baseband sample, and n is the time-sample index [3]. Each signal example contains 128 time samples. The complex signal is converted into a two-column matrix:

$$\mathrm{X} \in \mathbb{R}^{128\times 2} \tag{2}$$

The first column contains I samples, while the second column contains Q samples:

$$\mathrm{X} = \begin{bmatrix} s_I[0]s_Q[0] \\ s_I[1]s_Q[1] \\ \vdots \quad\quad \vdots \\ s_I[127]s_Q[127] \end{bmatrix} \in \mathbb{R}^{128\times 2} \tag{3}$$

The implemented TensorFlow/Keras models accept mini-batches of shape (N,128,2) and predict one of 11 modulation classes using a soft max output layer. Where N is the number of received signal examples.

B) Target Modulation Formats

The system classifies 11 target modulation formats, representing both digital and analog types:

TABLE I: THE MODULATION TYPES.

Class	Type	Construction
8PSK	Digital	Phase is changed into 8 levels separated by π/4.
AM-DSB	Analog	The envelope of the high-frequency carrier directly shows the analog message signal m(t).
AM-SSB	Analog	Amplitude modulation where either the upper or the lower sideband is filtered out to save power and bandwidth.
BPSK	Digital	Phase is changed into 2 levels, 0 and π.

CPFSK	Digital	Frequency shift keying is employed and the signal phase is kept continuous avoiding phase jumps and minimizing high-frequency spectral side-lobes.
GFSK	Digital	The digital data pulse is filtered with a Gaussian filter before frequency modulation. This limits bandwidth.
PAM4	Digital	The amplitude of the, in-phase component is changed into 4 distinct levels (−3,−1,1,3) while the quadrature componentremains at zero.
QAM16/QAM64	Digital	Both amplitude and phaseare changed to create grids of 16 and 64 points in the complex plane.
QPSK	Digital	Phase is changed into 4 levels separated byπ/2.
WBFM	Analog	The frequency of the carrier changes proportionally to the amplitude of m(t), with a high frequency deviation ratio.

C) CNN Model

The CNN model uses three convolutional layers to learn features from the I/Q sequence [13][14]. The layer-by-layer details of the1D-CNN configurations are summarized in Table2. The corresponding architectural flow is illustrated in Fig. 3.

TABLE II: LAYERCONFIGURATIONSANDSHAPESFORTHE1D-CNN MODELS.

Layer Type	Parameters	Output (CNN-50)	Output (CNN-40)
Input	–	128×2	128×2
ZeroPadding1D	padding=4	136×2	136×2
Conv1D	K=8,Stride=1,ReLU	129×50	129×40
MaxPooling1D	poolsize=2	64×50	64×40
Conv1D	K=8,Stride=1,ReLU	57×50	57×40
MaxPooling1D	poolsize=2	28×50	28×40
Conv1D	K=4,Stride=1,ReLU	25×50	25×40
Dropout	rate=0.6	25×50	25×40
MaxPooling1D	poolsize=2	12×50	12×40
Flatten	-	600	480
Dense	ReLU	70	70
Dense(Output)	Softmax	11	11

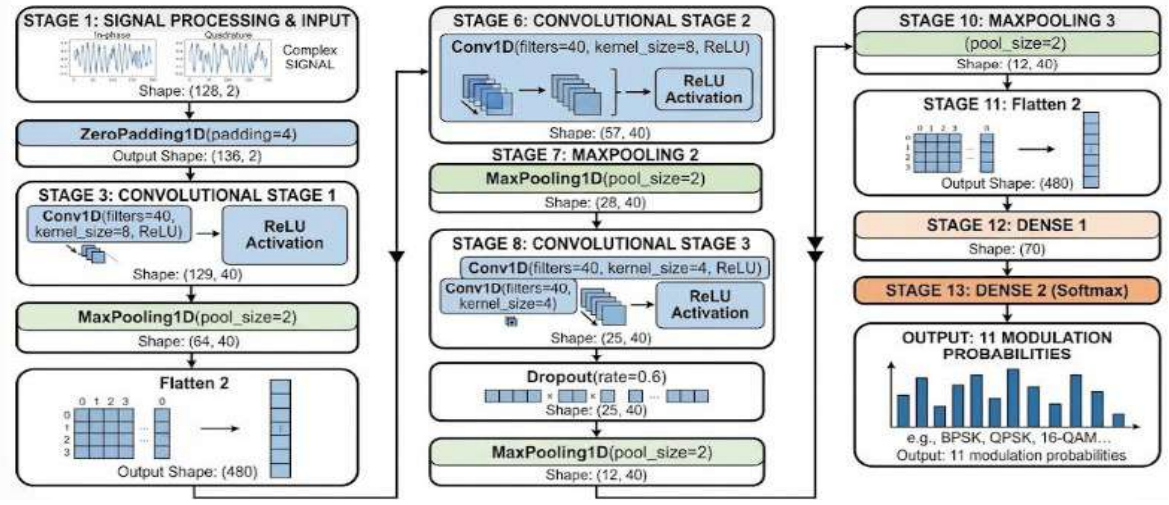


Fig. 3 CNN architecture for I/Q signal classification.

The sequential structure of CNN is as follows: The architecture starts with a Zero Padding Layer that adds 4 units of padding to change the input size from 128×2 to 136×2. The first processing step is the First Convolutional Layer (Conv1D-1) with F_1 filters of kernel size 8 and ReLU activation, which produces an output shape of $129\times F_1$. Next comes the First Max Pooling Layer, which reduces the sequence length to $64\times F_1$. Then a Second Convolutional Layer uses F_2 filters of a kernel size 8 and ReLU giving a size of $57\times F_2$. Another Max Pooling Layer further decreases the size to $28\times F_2$. The Third Convolutional Layer applies F_3 filters with a kernel size of 4 producing an output of $25\times F_3$ followed by a Dropout Layer at a rate of 0.6 to stop overfitting. Then the Third Max Pooling Layer, which results in a size of $12\times F_3$. The output is flattened into a vector of length $12\cdot F_3$ before it is handled by the First Dense Layer (FC-1) with 70 units and ReLU activation. Finally the Output Dense Layer (FC-2) uses a softmax activation function with 11 units giving a probability distribution, for eleven modulation types.

We explore two configurations for this model:

- CNN-40 (Model 1): Setting $F_1 = F_2 = F_3 = 40$ filters. The flattened vector has a length of 480, resulting in a total parameter size of 54,411.
- CNN-50 (Model 2): Setting $F_1 = F_2 = F_3 = 50$ filters. The flattened vector has a length of 600, resulting in a total parameter size of 73,730 (or 73,801) .

D) Hybrid CLDNN Model

CLDNN combines convolution and recurrent processing. The layer configurations of the hybrid CLDNN are summarized in Table 3, and the architectural flow is shown in Fig. 4.

TABLE III: LAYER CONFIGURATIONS AND SHAPES FOR THE HYBRID CLDNN MODELS

Layer Type	Parameters	Output (CNN-50)	Output (CNN-40)
Input	–	128×2	128×2
Conv1D	K=8,Stride=1,ReLU	121×64	121×54
MaxPooling1D	Poolsize=2	60×64	60×54
LSTM 1	units = CCL, return seq = True	60×64	60×54
Dropout	rate = 0.4	60×64	60×54
LSTM 2	units = CCL, return seq = True	60×64	60×54
Dropout	rate=0.4	60×64	60×54
Flatten	-	3,840	3,240
Dense(Output)	Softmax	11	11

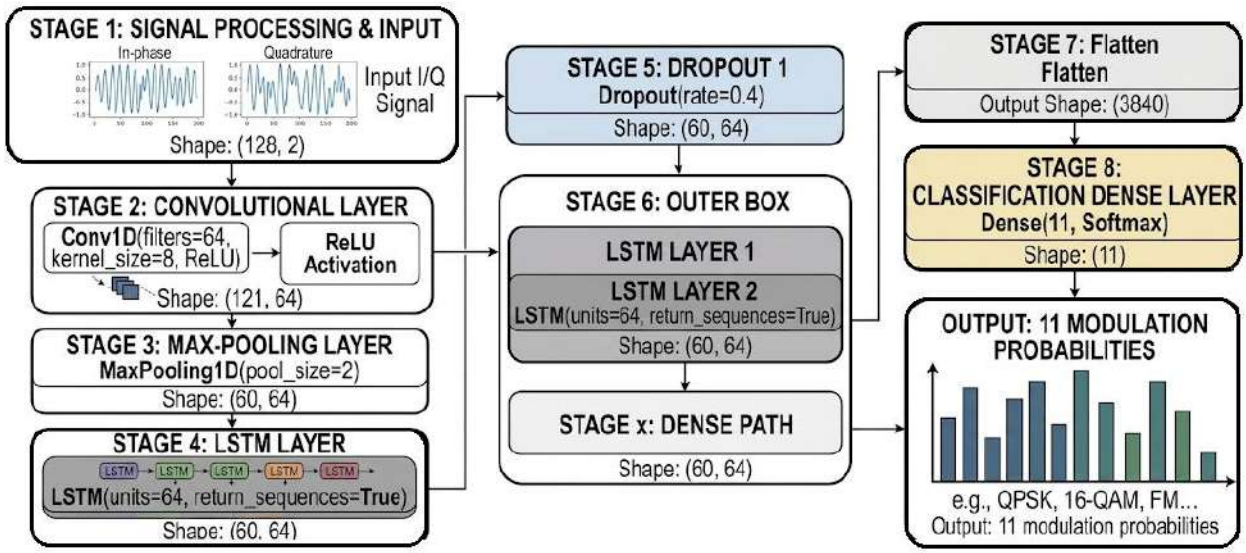


Fig. 4 CNN architecture for I/Q signal classification.

The sequential architecture of CLDNN consists of: The network starts with a Convolutional Front-End that uses F_c filters of a kernel size 8. This part takes an input that's 128×2 and turns it into a feature space that is $121\times F_c$. Then a Max Pooling layer reduces the size to $60\times F_c$. The first LSTM layer has L_1 hidden units. It returns sequences. This layer gives an output that's $60\times L_1$ and then a dropout layer with 0.4 is applied to it. The second LSTM layer has L_2 hidden units. It also returns sequences. This layer gives an output that's $60\times L_2$ and then another dropout layer with 0.4 is used. The network ends with a Flatten layer that makes the tensor, into one vector that has a length of $60\cdot L_2$. Finally there is an output layer that uses Softmax activation to predict 11 target classes. We compare two configurations for the CLDNN model:

- CLDNN-54 (Model 1): Setting $F_c = L_1 = L_2 = 54$. The flattened vector has a length of 3240, leading to a total parameter size of 83,657.
- CLDNN-64 (Model 2): Setting $F_c = L_1 = L_2 = 64$. The flattened vector has a length of 3480, leading to a total parameter size of 105,546 (or 109,387).

The comparative parameters and architectural characteristics of all four models under test aresummarized in Table4.

TABLE IV: MODEL PARAMETERS AND LAYER CONFIGURATIONS.

Model Configuration	Conv Filters / LSTM Units	Flatten Shape	Total Parameters
CNN-40	40	480	54,411
CNN-50	50	600	73,730
CLDNN-54	54	3,240	83,657
CLDNN-64	64	3,840	105,546

IV. Experimental Setup and Practical Simulation

To evaluate the models fairly, the training setup is kept the same for every test, to ensure that performance differences are produced from network architectures.

A) Training Parameters and Partitioning

The dataset is processed using a stratified splitting strategy to ensure an even distribution of classes across all subsets. The dataset split is configured as follows:

- **Train-Test Split**: 80% of the data is used for training, and the remaining 20% is reserved for final evaluation.
- **Validation Split**: During the training process, 15% of the training split is extracted as validation data to monitor the network's learning curve and prevent overfitting.

The training parameters configured in Keras are:

- **Batch Size**: 128 samples per step. This provides a clean balance between stochastic gradients and memory usage.
- **Epochs**: 50 complete passes over the dataset.
- **Optimizer**: Adam with a configured learning rate of 0.0007.
- **Loss Function**: Categorical Cross-Entropy, given the multi-class classification goal.

B) Evaluation Metrics

The main metric we used was classification accuracy, which's simply the percentage of signals that the model got right at each SNR level. This number tells us away how well the classifier can spot different modulation schemes when the noise changes.

To dig deeper into how the model behaves we made confusion matrices for six SNR values: −10 dB, −4 dB, 0 dB, 4 dB 10 dB and 16 dB. These tables show where the model mixes up modulation formats and which ones it keeps getting wrong.

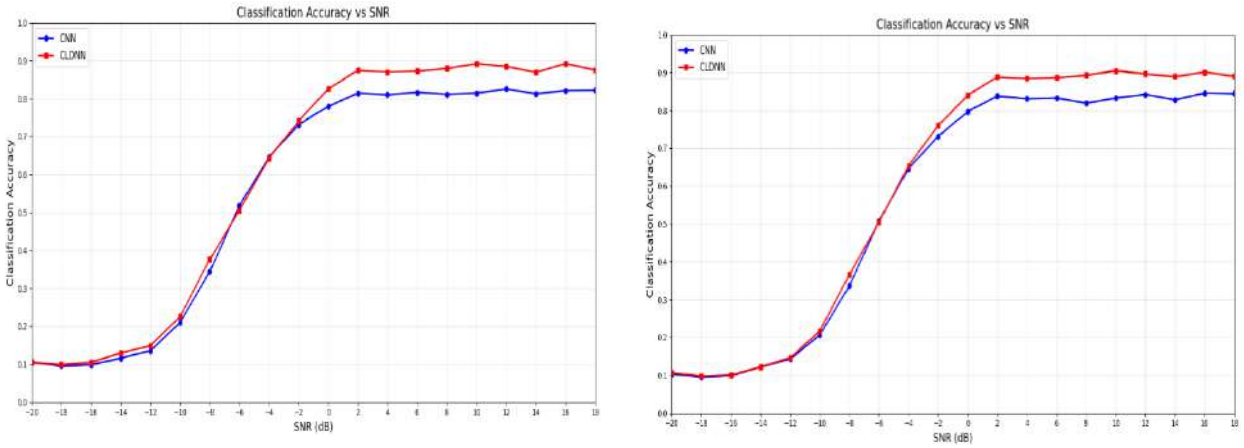
Finally, we drew accuracy-versus-SNR curves for both the CNN and the CLDNN. These graphs show how classification accuracy rises as signal quality improves and let us compare how robust each deep learning model is, across the SNR span.

V. Results and Discussion

Following the completed training, we analyze the over all performance through accuracy curves and detailed confusion matrices under varying channel conditions.

A) Accuracy vs. SNR Curve Analysis

The main performance of modulation classification can be captured by presenting the classification accuracyplotted against the Signal-to-Noise Ratio (SNR). To evaluate how different network capacities handle noise, we present and compare the Accuracy vs. SNR curves of both evaluated models: Model 1 (comprising the scaled-down CNN-40 and CLDNN-54) and Model 2 (comprising the standard-capacity CNN-50 and CLDNN-64)as shown in Fig. 5.



(a) Model 1 (CNN-40 vs. CLDNN-54) (b) Model 2 (CNN-50 vs. CLDNN-64)

Fig. 5 : Classification accuracy comparison between 1D CNN and hybrid CLDNN architectures for both Model configurations across the complete SNR range.

Looking at the curves in Fig.5, we can divide the performance of both systems into three distinct physical zones:

1- Noise-Dominated Zone (−20 dB to −12 dB): In this range, the noise power is large compared to the signal. Both models perform close to the random guess threshold of ≈ 9.09% (one out of 11 classes). The features of the signals are completely buried in Gaussian noise, making learning mathematically impossible for both configurations.

2- Transition Zone (−10 dB to 0 dB): Here, as the noise level drops, the accuracy climbs rapidly. This is where the models begin to extract meaningful patterns. In both figures, the CLDNN curve rises slightly faster and more robustly than the CNN. The spatial-temporal modeling of CLDNN allows it to recover weak signals (especially phase-modulated ones) better under high noise, demonstrating the value of sequential LSTM memory cells.

3- High-SNR Zone (> 0 dB): In the clean-signal zone, the performance difference becomes clear.
   - For Model 1 (Figure 5a), the 1D-CNN (CNN-40) plateaus at approximately 81.5% accuracy, while the hybrid CLDNN (CLDNN-54) reaches a peak accuracy of approximately 88%.
   - For Model 2 (Figure 3b), when model capacity is increased, the 1D-CNN (CNN-50) plateaus at approximately 82.0% accuracy, but the standard hybrid CLDNN (CLDNN-64) reaches a peak accuracy of approximately 89.5%.

A decrease in the size of parameter capacity of CNN and CLDNN models by around 25% reduces their complexity greatly without losing more than 1% in terms of peak classification capability. Hybrid CLDNN performs on average 7% better than only spatial CNN.

This is because the convolutional layer extracts only local features; it is able to detect changes in the local phase or the frequency but does not have a memory component to be aware of the total history of the temporal event. On the other hand, the CLDNN is able to trace the temporal trajectory

of the waveform and can therefore tell apart two complex schemes such as QAM16 and QAM64 that are represented on the same spatial grid.

B) Confusion-Matrix Analysis

The confusion matrices provide a more detailed view of the classification behavior than overall accuracy alone. In the following discussion, the diagonal elements represent correctly classified modulation samples, while the off-diagonal elements indicate confusion between modulation classes. The matrices are normalized by the number of samples in each true class; therefore, values close to one on the diagonal indicate strong recognition performance.

we inspect the exact confusion matrices generated for Model 1 and Model 2 across four critical operating points: low SNR (−10 dB), intermediate transition SNR (0 dB), high SNR (10 dB), and clean-signal plateau (16 dB) as shown in Fig. 6, 7,8,and 9.

1) Model 1 Confusion Matrix Analysis (CNN-40 vs. CLDNN-54)

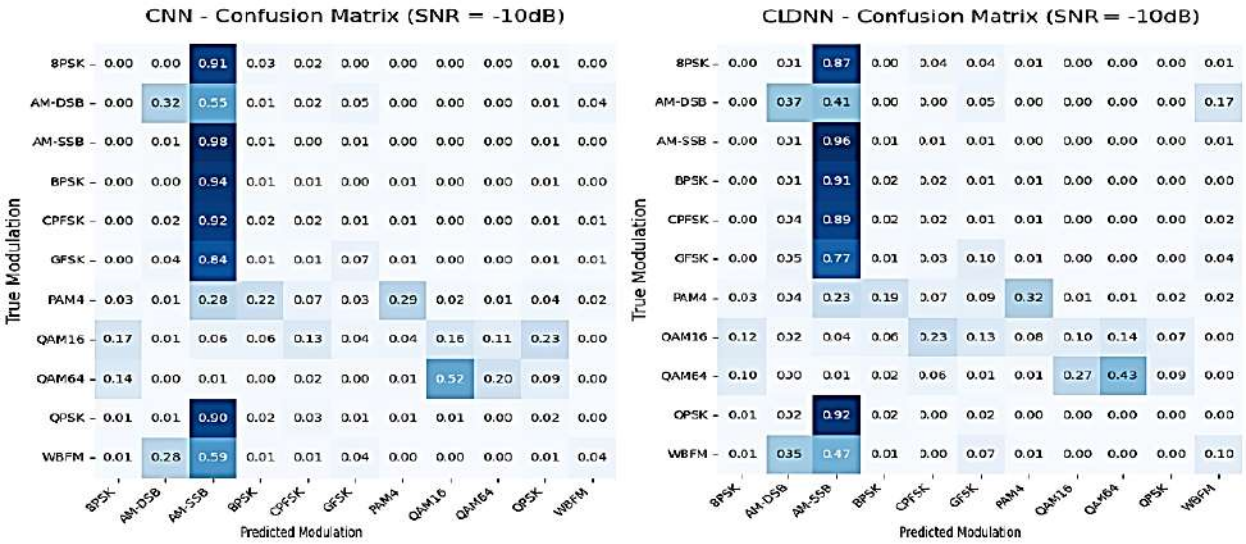


Fig. 6: Model 1 (CNN-40 vs. CLDNN-54) Confusion Matrices at SNR = −10 dB.

Both the CLDNN and the CNN models exhibit poor performance at SNR of -10 dB because of noise masking, making the classification difficult. The two models exhibit a bias for AM-SSB, where signals such as 8PSK and QPSK are misclassified into AM-SSB with an accuracy of over 77%. The CLDNN model retains the higher order QAM signals more accurately, and the model is able to classify 43% of QAM64 signals, as opposed to the CNN's 20%. However, CNN classifies PAM4 and QAM16 more accurately.

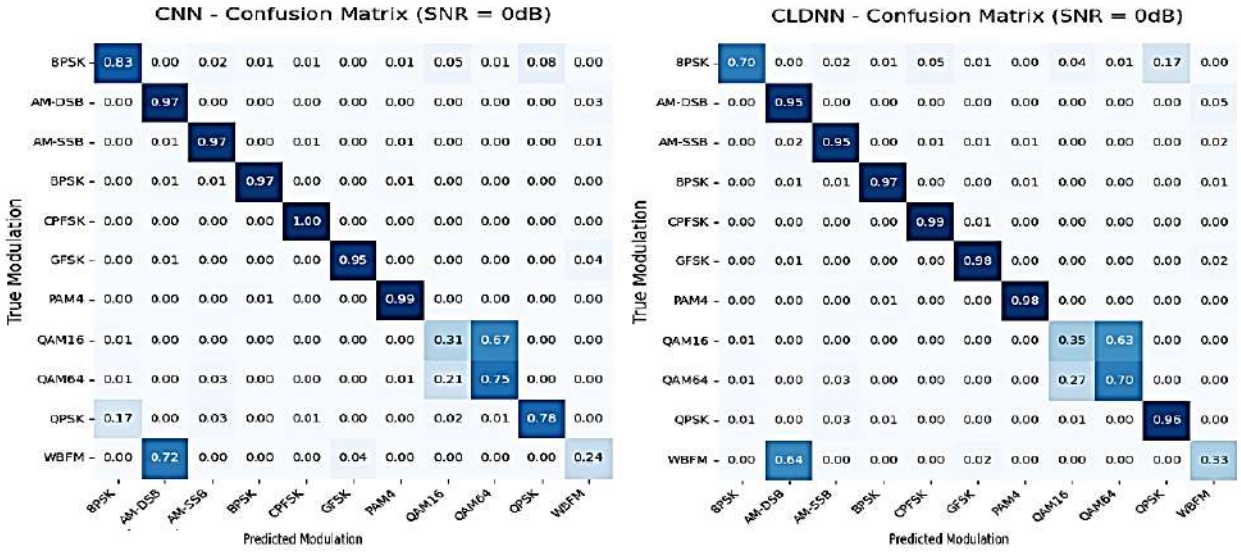


Fig. 7: Model 1 (CNN-40 vs. CLDNN-54) Confusion Matrices at SNR = 0 dB.

When SNR is 0 dB, CLDNN exhibits enhanced stability in phase sensitive modulation in comparison with CNN, giving more than 95% recognition in case of AM-DSB, AM-SSB, BPSK, CPFSK, GFSK, and PAM4. CLDNN successfully mitigates the off diagonal degradation in QPSK and surpasses CNN in cross confusion with 8PSK. Both networks are struggling with high order QAM phase amplitude overlap and analog feature extraction issues especially with QAM16, QAM64, and WBFM to AM-DSB.

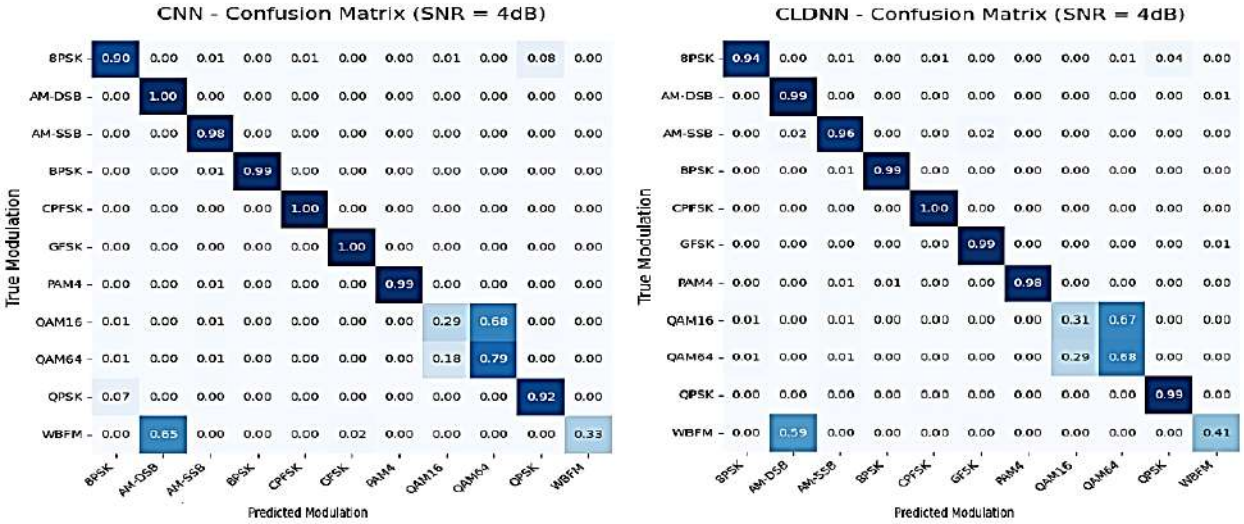


Fig. 8: Model 1 (CNN-40 vs. CLDNN-54) Confusion Matrices at SNR = 10dB.

Both CLDNN and CNN exhibit excellent convergence capabilities at SNR = 10 dB, with CLDNN exhibiting high accuracies across many modulation schemes such as 8PSK and AM-DSB, which stand at 97% and 100% respectively. The CLDNN enhances QPSK accuracy from 92% to 99%, through removal of misclassifications. Nevertheless, there is presence of consistent misclassification errors among higher order QAM classes, QAM16 and QAM64, as well as substantial analog carrier interference among WBFM.

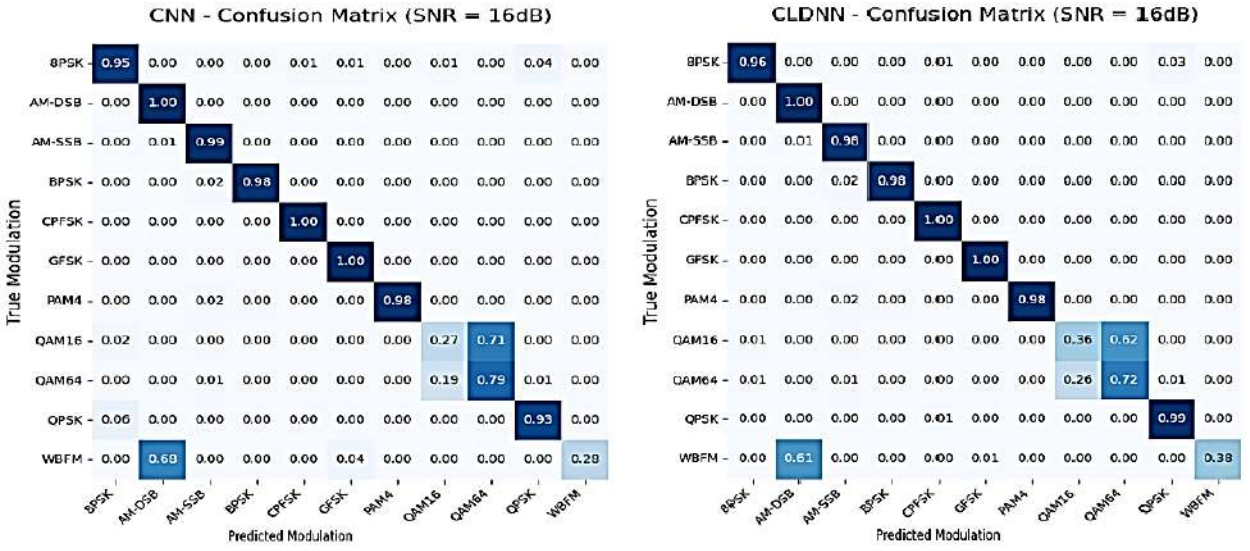


Fig. 9: Model 1 (CNN-40 vs. CLDNN-54) Confusion Matrices at SNR =16 dB.

When the SNR is 16 dB, both the models have very high levels of convergence, but the CLDNN model has a perfect level of classification of various modulations such as 8PSK, which is classified with a 96% rate of accuracy while 100% accuracy was attained in classifying AM-DSB. The CLDNN model performs better than the CNN model since there is no confusion in classifying QPSK and better accuracy in QAM16 and WBFM.

2) Model 2 Confusion Matrix Analysis (CNN-50 vs. CLDNN-64)

We present the confusion matrices of the standard-capacity Model 2 at the exact same sequence of operating points in Fig.10, 11, 12, and 13.

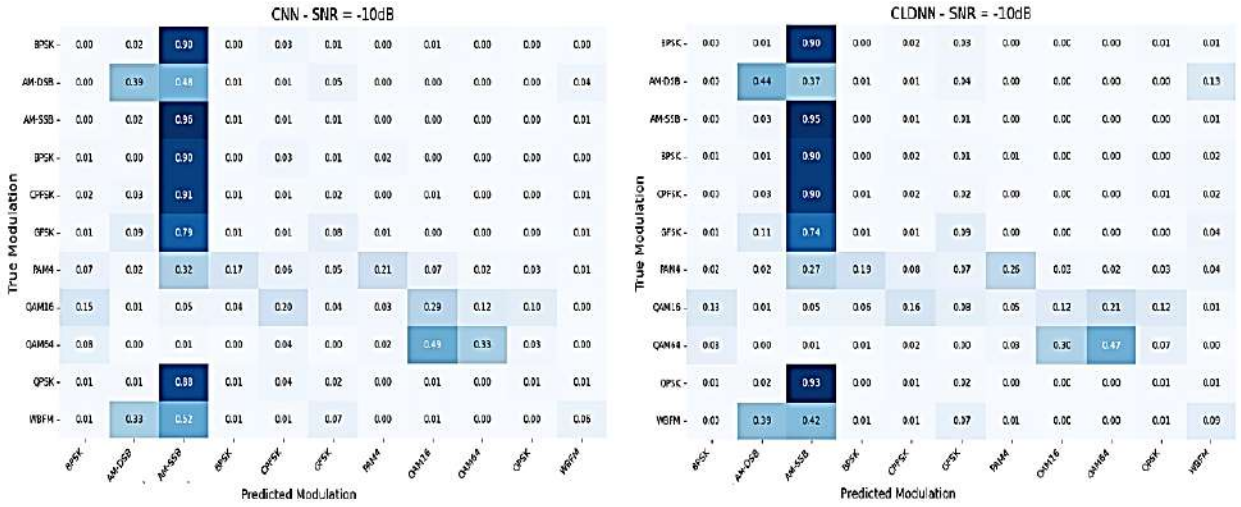


Fig. 10: Model 2 (CNN-50 vs. CLDNN-64) Confusion Matrices at SNR = -10dB.

At SNR = -10 dB, both models face extreme noise deterioration, giving rise to off-diagonal dispersal along with diagonal disappearance. One prominent form of failure entails the transformation of PSK and FSK to AM-SSB, with 8PSK, BPSK, CPFSK, and QPSK having more than 88% of mis-classified data in the respective column. Although CLDNN shows certain robustness regarding the accuracy of QAM64 (47%), as opposed to that of CNN (33%), the former has superior accuracy for QAM16 (29% vs. 12%).

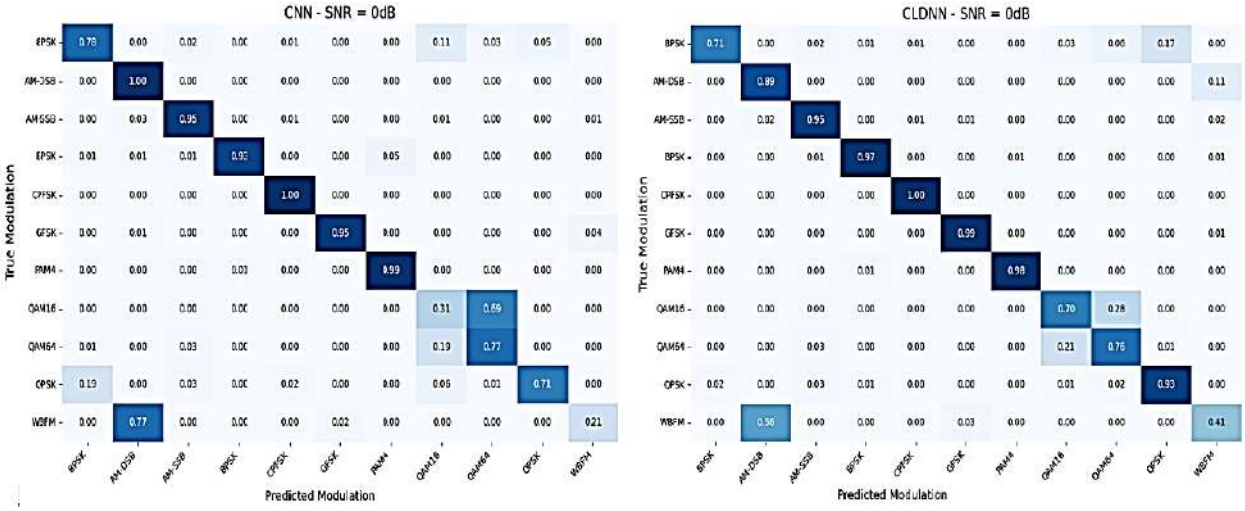


Fig. 11: Model 2 (CNN-50 vs. CLDNN-64) Confusion Matrices at SNR = 0dB.

When SNR = 0 dB, CLDNN shows better diagonal clustering than CNN, producing perfect classification accuracy greater than 95% in AM-SSB (95%), BPSK (97%), CPFSK (100%), GFSK (99%), and PAM4 (98%). In addition, CLDNN reduces off-diagonal clustering in QPSK (93% as

against 71%) and QAM16 (70% as against 31%), while CNN suffers from a lot of confusion between classes, specifically in classifying 69% of QAM16 as QAM64.

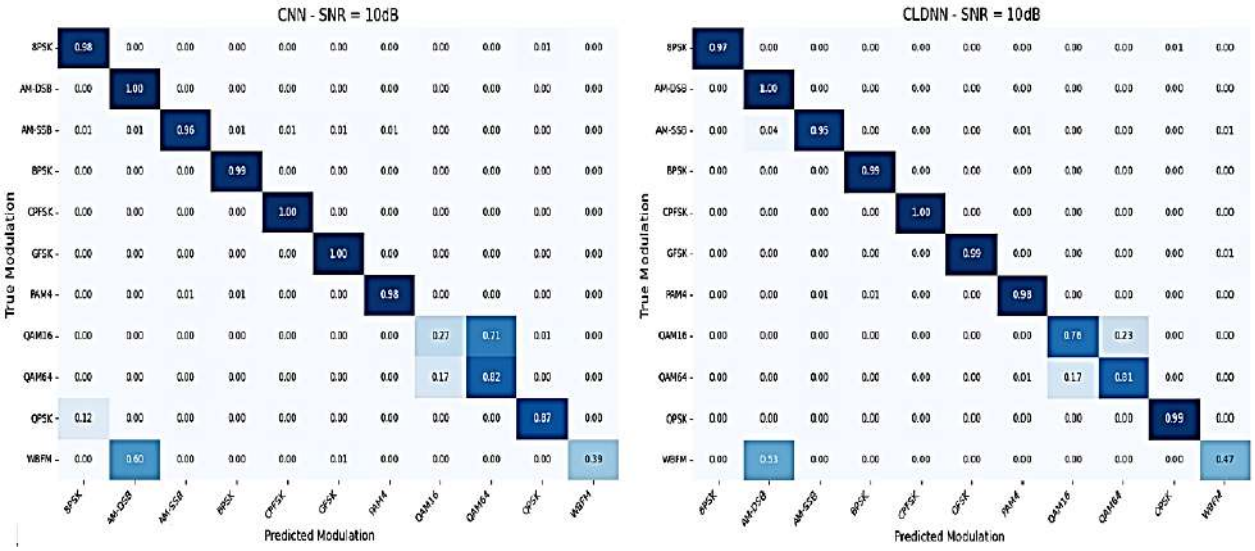


Fig. 12: Model 2 (CNN-50 vs. CLDNN-64) Confusion Matrices at SNR = 10dB.

With an SNR of 10 dB, both CLDNN and CNN networks have shown better convergence results where CLDNN shows more than 95% accurate results for all digital signals where 8PSK gives 97% accuracy and AM-SSB & CPFSK show 100% accuracy. Moreover, the CLDNN network shows a marked improvement in QAM16 signal accuracy from 27% to 76%, resolving the issue of QAM constellation blur that is present in the CNN network. In addition, the CLDNN network resolves the issue of QPSK-8PSK ambiguity and reduces WBFM crosstalk from 47% to 39%.

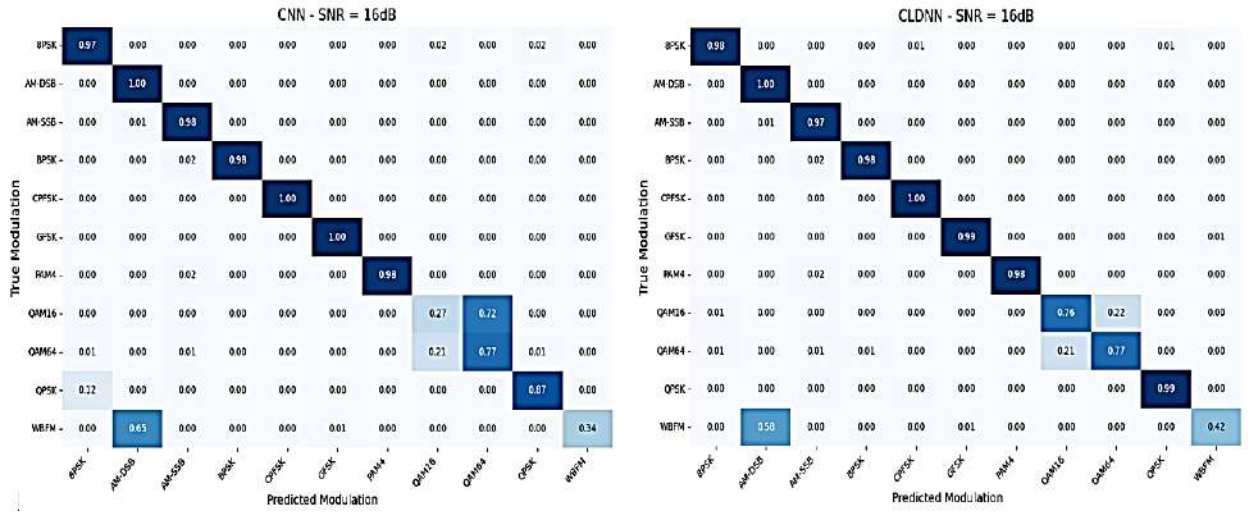


Fig. 13: Model 2 (CNN-50 vs. CLDNN-64) Confusion Matrices at SNR =16 dB.

Both the networks provide perfect performance for all digital classes such as 8PSK and BPSK at SNR = 16 dB with more than 97% diagonal classification. On the other hand, CLDNN is able to perform better than the CNN through elimination of any confusion related to off-diagonal, including distinguishing between QPSK-8PSK (99% against 87%) and QAM16 accuracy (76% against 27%).

C) Cross-Model and Cross-Architectural Comparison

The results show that CLDNN consistently outperforms CNN under both width settings, while revealing clear trade-offs between accuracy, model size, and temporal modeling.

CNN models have low accuracy rates. CLDNN models have high accuracy rates of about 88% for CLDNN-54 and 89.5% for CLDNN-64 improving the accuracy rates by about 6.5-7.5 percentage points. Increasing the width of CLDNN leads to better accuracy gains than CNN.

The confusion matrices show that CLDNN performs well on modulation classes that require temporal reasoning and can significantly improve the recognition rates of QPSK and QAM16 at different dB levels.

The numbers of parameters for CNN-40, CNN-50, CLDNN-54, and CLDNN-64 are 54,411, 73,730, 83,657, and 105,546, respectively. In contrast with CLDNN-64, CLDNN-54 uses 20.7% fewer parameters, keeping the peak accuracy reduction at just 1.5. Hence, there is a nice compromise between accuracy and complexity. Although CNN-40 has 48.5% fewer parameters than CLDNN-64, its peak accuracy is decreased by 8.0, making it a better option for limited memory applications.

CLDNN models require more computation than CNN models, because of the recurrent nature of the Long-Short Term Memory network. As a result, CLDNN-54 is a suitable model for use in applications that have limited computational resources. The CNN models efficiently learned the local I/Q patterns, whereas the CLDNN models benefited from the ability to perform temporal modeling through the LSTM layers.

VI. Conclusion and Future Work

A) Conclusion

This paper presented an in-depth comparative analysis of 1D-CNN and hybrid CLDNN models used in Automatic Modulation Classification (AMC). Based on our comparison using standardized settings, the hybrid CLDNN (reaching an optimum accuracy of 88%-89.5%) is proven to be superior to a pure convolutional 1D-CNN (having an optimum accuracy of 82%). The use of local convolutions as feature extractors coupled with temporal tracking memory of LSTM blocks is necessary in extracting the phase space trajectory of complicated communication methods. In addition to the above, we carried out an ablation study of different sizes of the two models. As observed from the result, reduction of the number of parameters (decreasing CNN filters from 50 to 40 and CLDNN units from 64 to 54), reducing the number of parameters by up to 20%-25%, leads to dramatic reductions in the total number of parameters with only a small decrease (around 1%-1.5%) in optimum classification accuracy.

B) Future Work

In the future, we plan to extend this work in three main directions:

1. **Transformer-Based Sequence Models:** Implementing self-attention mechanisms (such as Vision Transformers adapted for 1D time-series) to capture extremely long-term phase interactions that might escape LSTMs .
2. **Non-Gaussian and Fading Channels:** Testing model robustness in harsh channels containing multipath Rayleigh/Rician fading and impulsive non-Gaussian noise, as well as carrier frequency and phase offsets .
3. **Hardware Deployment:** Quantizing the scaled-down CLDNN-54 model to 8-bit integers (INT8) and compiling it to run on embedded FPGA or ARM platforms for real-time edge spectrum analysis, complemented by over-the-air SDR evaluation.

REFERENCES

[1] Thakur, A C S, & Imtiaz M. (2026). Deep learning for automatic modulation classification: A review. Electronics, 15(10), Article 2163. https://doi.org/10.3390/electronics15102163

[2] Padhya D, Acharya K, Dahal B K, &Kshatri D B. (2025). CNN-LSTM hybrid architecture for over-the-air automatic modulation classification using SDR. arXiv. https://arxiv.org/abs/2511.21040

[3] O'Shea T J, Corgan J, & Clancy T C. (2016). Convolutional radio modulation recognition networks. In International Conference on Engineering Applications of Neural Networks (pp. 213–226). Springer. https://doi.org/10.1007/978-3-319-44160-3_16

[4] O'Shea T J, Roy T, & Clancy T C. (2018). Over-the-air deep learning based radio signal classification. IEEE Journal of Selected Topics in Signal Processing, 12(1), 168–179. https://doi.org/10.1109/JSTSP.2018.2797022

[5] Wang Z, Zhang W, Zhao Z, Tang P, & Zhang Z. (2024). Robust automatic modulation classification via a lightweight temporal hybrid neural network. Sensors, 24(24), Article 7908. https://doi.org/10.3390/s24247908

[6] Zhang Z, Luo H, Wang C, Gan C, & Xiang Y. (2020). Automatic modulation classification using CNN-LSTM based dual-stream structure. IEEE Transactions on Vehicular Technology, 69(11), 13521–13531. https://doi.org/10.1109/TVT.2020.3030018

[7] Abdulkarem A M, Abedi F, Ghanimi H M A, Kumar S, Al-AzzawiW K, Abbas A H, Abosinnee A S, Almaameri I M, &Alkhayyat A. (2022). Robust automatic modulation classification using convolutional deep neural network based on scalogram information. Computers, 11(11), Article 162. https://doi.org/10.3390/computers11110162

[8] Rajendran S, Meert W, Giustiniano D, Lenders V, &Pollin S. (2018). Deep learning models for wireless signal classification with distributed low-cost spectrum sensors. IEEE Transactions on Cognitive Communications and Networking, 4(3), 433–445. https://doi.org/10.1109/TCCN.2018.2835460

[9] Abd-Elaziz O F, Abdalla M, &Elsayed R A. (2023). Deep learning-based automatic modulation classification using robust CNN architecture for cognitive radio networks. Sensors, 23(23), Article 9467. https://doi.org/10.3390/s23239467

[10] Tekbıyık K, Ekti A R, Görçin A, Kurt G K, &Keçeci C. (2020). Robust and fast automatic modulation classification with CNN under multipath fading channels. In 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring) (pp. 1–5). IEEE. https://doi.org/10.1109/VTC2020-Spring48590.2020.9129188

[11] Zhu Z, & Nandi A. K. (2015). Automatic modulation classification: Principles, algorithms and applications. John Wiley & Sons

[12] DeepSig, "Datasets: RADIOML 2016.10A," historical dataset description and download page. https://www.deepsig.ai/datasets/.

[13] Abdou TS, Elghanemi MS, Dakhil AA, Khaleel M. Artificial neural network-based Active Noise Cancellation of ambulance. مجلة شمال إفريقيا للنشر العلمي (NAJSP). 2025 Jul 19:63-70.

[14] Wu T, Zhu Q, Mao R, Hu C, & Wei S. (2026). CAIC-Net: Robust radio modulation classification via unified dynamic cross-attention and cross-signal-to-noise ratio contrastive learning. Sensors, 26(3), Article 756. https://doi.org/10.3390/s26030756

[15] Xu Y. (2025). AMC-Transformer: Automatic Modulation Classification based on Enhanced Attention Model. Infocommunications Journal, 17(4), 32–40. https://doi.org/10.36244/ICJ.2025.4.5

[16] Sahu PC, Prasad B, Dash R, Muduli D, Kollie DS Jr, Priyadarshi R, Ranjan R. Cloud-enabled automatic modulation classification using deep feature fusion and Moth-Flame Optimized ELM approach. Scientific Reports. 2025 Dec:30753.

BIOGRAPHIES OF AUTHORS

Fathia Abduslam Hussain received the Diploma degree in Communication Engineering from College of Electronic Technology, Libya, in 1999, and the M.Sc. degree in Communication Engineering from Libyan Academy for Postgraduate Studies, Libya, in 2016. She is currently a Lecturer with the Department of Electrical and Electronic Engineering, Higher Institute of Engineering Techniques, Tripoli, Libya. Her research interests include AI, Mobile System, and Antenna. She has contributed to various scientific publications and academic activities.

Abdualteef Ahmed Al Shami received the Diploma degree in Communication Engineering from Higher Institute of Engineering Techniques, Libya, in 2026. He is currently a student with the Department of Electrical and Electronic Engineering, Higher Institute of Engineering Techniques, Tripoli, Libya. His research interests include AI.

Mahmoud Abdulnasir Owayli received the Diploma degree in Communication Engineering from Higher Institute of Engineering Techniques, Libya, in 2026. He is currently a student with the Department of Electrical and Electronic Engineering, Higher Institute of Engineering Techniques, Tripoli, Libya. His research interests include AI.

Dr. Tarek Abdou received his first degree in Electronic Engineering from the Higher Institute of Engineering Technology, Tripoli, Libya, in 1999, and the M.Sc. degree in Telecommunications from Sheffield Hallam University, United Kingdom, in 2015. He received the Ph.D. degree in Millimeter-Wave Engineering and Antenna Propagation from the University of Sheffield, United Kingdom, in 2023. He is currently an academic staff member with the Department of Electrical and Electronic Engineering, Communications Group, Higher Institute of Engineering Technology, Tripoli, Libya. His research interests include millimeter-wave antennas, antenna design, wireless communication systems, electromagnetic wave propagation, antenna propagation, and artificial intelligence (AI). He has contributed to scientific publications and academic activities.