# Automatic Modulation Classification using Shallow CNNs

*A*

*Thesis Submitted*

*in Partial Fulfilment of the Requirements*

*for the Degree of*

**DOCTOR OF PHILOSOPHY**

**By**


**Dileep P**

Roll No. 156102033



Department of Electronics and Electrical Engineering

Indian Institute of Technology Guwahati

Guwahati - 781 039, INDIA.


June, 2026

TH-4097_156102033

TH-4097_156102033

**Dedicated to**

**Jomy**

TH-4097_156102033

TH-4097_156102033

# Certificate

This is to certify that the thesis entitled "Automatic Modulation Classification Using Shallow CNNs", submitted by Dileep P (156102033), a research scholar in the Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati, for the award of the degree of Doctor of Philosophy, is a record of an original research work carried out by him under our supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the institute and in my opinion has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dated:

Place: Guwahati

Dr. A. Rajesh

Dept. of Electronics and Electrical Engg.

Indian Institute of Technology Guwahati

Guwahati - 781039, Assam, India.

Dated:

Place: Guwahati

Prof. P. K. Bora

Dept. of Electronics and Electrical Engg.

Indian Institute of Technology Guwahati

Guwahati - 781039, Assam, India.

TH-4097_156102033

TH-4097_156102033

# Declaration

I declare that this written submission represents my ideas in my own words, and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Dated:20$^{th}$ June, 2026

Place: Guwahati

**Dileep P**

**Roll No.: 156102033**

TH-4097_156102033

TH-4097_156102033

# Acknowledgements

This thesis would not have been possible without the invaluable guidance, encouragement, and support of my supervisors, Prof. Prabin Kumar Bora and Dr. A. Rajesh. I express my deepest gratitude to them for their continuous mentorship, insightful advice, and unwavering patience throughout the course of my research.

I also wish to extend my heartfelt appreciation to the members of my Doctoral Committee — Prof. Rohit Sinha, Prof. Ratnajit Bhattacharjee, and Dr. Salil Kashyap — for their constructive suggestions and critical feedback, which have been instrumental in shaping and improving the quality of this work.

I would also like to thank all my friends who stood by me through the long course of my doctoral journey, offering both academic assistance and much-needed mental support during my extended stay at the institute. Their companionship and encouragement made this journey truly meaningful and enjoyable. A special mention goes to Dr. Dibyajyoti Das, a Ph.D. graduate from the same department, for his valuable discussions, constant encouragement, and generous help that significantly contributed to my research and personal growth.

*(Dileep P)*

TH-4097_156102033

*"The important thing in science is not so much to obtain new facts as to discover new ways of thinking about them."*

William Lawrence Bragg

TH-4097_156102033

# List of Acronyms

| | |
|---|---|
| AMC | Automatic Modulation Classification |
| LB | Likelihood Based |
| FB | Feature Based |
| HOM | Higher Order Moment |
| DL | Deep Learning |
| CNN | Convolutional Neural Network |
| HOC | Higher Order Cumulant |
| MGF | Moment Generating Function |
| CGF | Cumulant Generating Function |
| FC | Fusion Center |
| ALRT | Average Likelihood Ratio Test |
| HLRT | Hybrid Likelihood Ratio Test |
| GLRT | Generalised Likelihood Ratio Test |
| SVM | Support Vector Machine |
| KNN | K Nearest Neighbour |
| LSTM | Long Short Term Memory |
| ZF | Zero Forcing |
| SCNN | Shallow CNN |
| IQ | In-phase and Quadrature |
| IoT | Internet of Things |
| HDF | Hard Decision Fusion |
| DrCNN | Dropout CNN |

TH-4097_156102033

| | |
|---|---|
| PCC | Percentage of Correct Classification |
| BN | Batch Normalisation |
| DDrCNN | Dense Dropout CNN |
| DBCNN | Dense Batch normalization CNN |
| ML | Maximum Likelihood |
| AWGN | Additive White Gaussian Noise |
| CAMC | Cooperative AMC |
| CSI | Channel State Information |
| ANN | Artificial Neural Network |
| DMR | Direct Modulation Recognition |

TH-4097_156102033

# Abstract

Automatic Modulation Classification (AMC) aims to identify the modulation scheme of an unknown communication signal. It is a widely researched area, with early studies primarily focusing on Single Input Single Output (SISO) systems.Two main approaches are widely adopted for AMC: the likelihood-based (LB) approach and the feature-based (FB) approach. The LB approach treats AMC as a multiple-hypothesis testing problem, where each of the $L$ composite hypotheses corresponds to a distinct modulation type. FB methods are more computationally efficient and require less prior information, relying instead on features such as higher-order moments (HOMs) and cumulants (HOCs) computed from the received signals.

The developments in deep learning (DL) algorithms for computer vision applications have opened the door to eliminate the need for handcrafted feature extraction in AMC. The method of representing complex baseband communication signals as two-dimensional inputs, separating the real and imaginary components into distinct channels enabled the use of standard convolutional neural network (CNN) architectures for signal classification. Subsequent studies have extended this approach using various DL models for AMC, consistently demonstrating the superiority of DL-based methods over traditional LB and FB classifiers in terms of classification accuracy and generalization. Though many results are reported on employing DL for AMC, several improvement to the existing classifiers and further areas where DL classifiers can be effectively applied were evident. The AMC task is generally deployed in resource-constrained systems, making it essential to minimize the size and computational complexity of the classifier design. This thesis addressed these issues in the following problems by focussing on shallow CNNs (SCNN):

1. The AMC of eight digital modulation schemes—BPSK, QPSK, 8PSK, GFSK, CPFSK,

TH-4097_156102033

PAM4, 16QAM, and 64QAM—over a Rician channel employing an SCNNs is addressed. A popular classifier, the dropout CNN (DrCNN), was able to classify six of these classes but not the QAM schemes. To address this limitation, a modified SCNN architecture called the dense layer dropout CNN (DDrCNN) is proposed to classify the full set of modulations while keeping the complexity low. The effects of dropout and batch normalization (BN) on this dataset are studied.

2. In communication systems, the availability of multiple receiver nodes helps to improve the classification accuracy through cooperation between these nodes. This thesis proposes a study on enhancing CNN classifier performance through cooperation between communicating nodes. A decision fusion method with a centralized architecture is selected for evaluation. Popular cooperation methods, such as voting and averaging-based decision making, are compared in the study..

3. The application of DL techniques for the AMC of MIMO signals remains limited in existing research. Most feature-based methods rely on channel estimation prior to classification. This thesis addresses the direct AMC of MIMO signals using a modified version of the shallow CNN, proposed for the single-input single-output signal classification. When trained with the in-phase and quadrature representation of all inputs from multiple receiver antennas, the classifier achieved less than 90 percent accuracy. To overcome this, a cooperative mechanism across receiving antennas was introduced, enhancing overall classification performance. Both decision-fusion and feature-fusion cooperation schemes were implemented, and the resulting performance was compared with other similar works reported in the literature, confirming the superior accuracy of the modified DDrCNN for AMC of MIMO signals over fading channels.

4. Cumulant-based AMC methods perform poorly for MIMO signals in rank-deficient channels such as keyhole channels, since accurate cumulant estimation depends on reliable channel state information (CSI). In keyhole conditions, the fading process causes complex mixing and distortion, leading to unreliable estimates. The proposed DL-based AMC

TH-4097_156102033

approach was evaluated on MIMO keyhole channel signals to test its robustness. The classifier developed using the modified DDrCNN architecture and trained on full-rank MIMO signals exhibited robust generalization, achieving satisfactory performance on keyhole channel conditions as well.

TH-4097_156102033

TH-4097_156102033

# Contents


**List of Acronyms** **xi**

**List of Figures** **xxi**

**List of Tables** **xxv**

**1 Introduction** **1**

1.1 General Definition of AMC Problem . . . . . 2

1.2 LB Method for AMC . . . . . 3

1.3 FB Method for AMC . . . . . 5

1.3.1 HOC based AMC . . . . . 5

1.3.2 Cyclic Cumulants based classifiers . . . . . 6

1.4 Deep Learning for AMC . . . . . 6

1.5 Existing Works on DL-based AMC . . . . . 7

1.6 AMC using shallow CNNs . . . . . 8

1.7 Datasets used in DL-AMC . . . . . 9

1.8 Motivation and Problem Formulation . . . . . 10

1.8.1 Thesis organisation . . . . . 12

**2 AMC of SISO signals using Shallow CNNs** **13**

2.1 Introduction . . . . . 14

2.2 Strategies to Mitigate Overfitting in CNNs . . . . . 15

2.2.1 Dropouts as a regulariser . . . . . 16

2.2.2 Batch Normalisation (BN) . . . . . 17

2.3 Proposed SCNN Classifiers for the AMC of 8 Digital Modulation Classes. . . . 19

2.3.1 DDrCNN for AMC . . . . . 19


TH-4097_156102033

2.3.2 DBCNN for AMC . . . . . . . . . . . . . . . . . . . . . . . . . 21
2.4 Implementation Details and Results . . . . . . . . . . . . . . . . . . . . . 21
2.4.1 Dataset . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
2.4.2 Selection of the CNN architecture for the AMC . . . . . . . . . . . . . 23
2.4.3 The effect of the size and representation of the signals in the dataset on the PCC . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 34
2.4.4 Performance comparison with DrCNN . . . . . . . . . . . . . . . . 34
2.5 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 37

**3 Cooperative Modulation Classification Employing SCNNs over Distributed Communication Networks 39**

3.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 40
3.2 Related Works on Cooperative AMC over Distributed Networks . . . . . . . . 42
3.3 System Model and Proposed Classifier . . . . . . . . . . . . . . . . . . . . 43
3.3.1 CNN used for classification . . . . . . . . . . . . . . . . . . . . . 44
3.3.2 CAMC over Distributed SISO Networks . . . . . . . . . . . . . . . . 44
3.4 Simulation Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . 46
3.4.1 Simulation results with signal size 2x128 . . . . . . . . . . . . . . . 47
3.4.2 Simulation results with signal size 2x64. . . . . . . . . . . . . . . . 49
3.5 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 52

**4 AMC of MIMO Signals using Shallow CNNs 53**

4.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 54
4.2 AMC of MIMO Signals Over Rayleigh Channel . . . . . . . . . . . . . . . . 56
4.2.1 MIMO Rayleigh Channel Model . . . . . . . . . . . . . . . . . . . 56
4.3 Proposed SCNN Architecture for the AMC of MIMO Signals . . . . . . . . . 58
4.3.1 CAMC of MIMO Signals for Performance Improvement . . . . . . . . 60
4.3.2 Other Competing CNNs Selected for Comparison . . . . . . . . . . . 61
4.4 Results and discussions . . . . . . . . . . . . . . . . . . . . . . . . . 62
4.4.1 Generation of the MIMO Dataset . . . . . . . . . . . . . . . . . . . 62

TH-4097_156102033

4.4.2 Performance evaluated on non-cooperative AMC dataset . . . . . . . . 63
4.4.3 Results for cooperation among multiple antenna signals . . . . . . . . 65
4.4.4 Classification results for classifier employing feature fusion. . . . . . . 69
4.5 AMC Performance of the Modified DDrCNN on MIMO Signals Over Keyhole Channel. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 70
4.5.1 MIMO Keyhole Channel . . . . . . . . . . . . . . . . . . . . . . . . 70
4.5.2 Various Training Scenarios Explored for the Modified DDrCNN . . . . 71
4.5.3 Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 73
4.5.3.1 Comparison of SCNN-based AMC over rank-deficient channel with traditional FB-AMC method . . . . . . . . . . . . . 78
4.6 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 79
**5 Conclusions and Future Works** **81**
5.1 Conclusions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 82
5.2 Scope for Future Works . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 84
**List of Publications** **88**
**Bibliography** **89**

TH-4097_156102033

TH-4097_156102033

# List of Figures


2.1 A schematic diagram of the existing DrCNN-based classifier for the AMC of eight modulation classes. . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
2.2 Dropout applied to a standard neural network . . . . . . . . . . . . . . . . . . 17
2.3 The process block diagram of a BN layer. . . . . . . . . . . . . . . . . . . . . 18
2.4 A schematic diagram of the DDrCNN-based classifier for the AMC of eight modulation schemes. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20
2.5 Effect of dropouts only in dense layer. . . . . . . . . . . . . . . . . . . . . . . 26
2.6 Training and validation loss over epochs (DDrCNN). . . . . . . . . . . . . . . 27
2.7 Performance of DDrCNN classifier for various SNRs compared with CNN1. . . 27
2.8 Confusion matrix of DDrCNN for various SNRs. . . . . . . . . . . . . . . . . 28
2.9 PCC plot for various intermediate CNNs listed in Table 2.4. . . . . . . . . . . . 30
2.10 PCC of DBCNN compared with CNN3 and CNN6. . . . . . . . . . . . . . . . 30
2.11 Training and validation loss for various CNNs . . . . . . . . . . . . . . . . . . 31
2.12 PCC performance of DDrCNN and DBCNN on RadioML dataset. . . . . . . . 32
2.13 PCC performance of DDrCNN for different lengths of signals. . . . . . . . . . 32
2.14 PCC performance of DDrCNN for Polar and IQ dataset. . . . . . . . . . . . . 33
2.15 Confusion matrix of DDrCNN for IQ data and Polar data. . . . . . . . . . . . . 33
2.16 Comparison of the Confusion matrix of DrCNN, DBCNN and DDrCNN. . . . 36
2.17 Performance comparison of DrCNN with proposed networks on the synthetic dataset. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 37

3.1 A distributed communication network with centralised fusion for decision co-operation. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 41


TH-4097_156102033

3.2 CNN-based system for CAMC over a distributed network. . . . . . . . . . . . 43
3.3 Architecture of CNN employed at each node of the distributed network. . . . . 44
3.4 Model training behaviour and classification performance for a single node (2×128 input format) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 48
3.5 PCC for CAMC with multiple SISO nodes for input size 2x128 . . . . . . . . . 49
3.6 Accuracy comparison of CAMC: Averaging vs. Voting (Input Size: 2×128) . . 50
3.7 Model training behaviour and classification performance for a single node (2×64 input format) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 51
3.8 PCC for CAMC with multiple SISO nodes for input size 2x64 . . . . . . . . . 52

4.1 The structure of CNN-based AMC of MIMO signals - non-cooperative method. 57
4.2 Structure of the modified DDrCNN. . . . . . . . . . . . . . . . . . . . . . . . 58
4.3 Schematic diagram of the AMC system employing decision fusion cooperation. 60
4.4 Schematic diagram of the AMC system employing feature fusion. . . . . . . . 61
4.5 Non-cooperative DL-AMC of MIMO signals over Rayleigh fading channel ($N_t$ = 2, $N_r$ = 4). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 64
4.6 PCC for Modified DDrCNN-based CAMC (decision fusion) over MIMO Rayleigh channel for the 5-modulation pool ($N_t$ = 2, $N_r$ = 4) . . . . . . . . . . . . . . 65
4.7 Confusion matrix of Modified DDrCNN at various SNRs on the MIMO dataset ($N_t$ = 2, $N_r$ = 4). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 66
4.8 Comparison of PCC for DL-based CAMC (decision fusion) over MIMO Rayleigh channel for the 5-modulation pool ($N_t$ = 2, $N_r$ = 4) . . . . . . . . . . . . . . 67
4.9 PCC for cooperative AMC over correlated MIMO channel ($N_t$ = 2, $N_r$ = 4) . . 68
4.10 PCC for decision fusion vs feature fusion over uncorrelated MIMO channel ($N_t$ = 2, $N_r$ = 4) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 69
4.11 Representation of a spatial keyhole channel. . . . . . . . . . . . . . . . . . . . 70
4.12 CNN classifier for signals over rank deficient channels. . . . . . . . . . . . . . 71
4.13 AMC employing the SCNN classifier trained with signals over keyhole channel 72

4.14 AMC employing the SCNN classifier trained with signals over a full rank channel. . . . 72
4.15 PCC for keyhole channels while the classifier was trained with similar signals . 74
4.16 PCC for keyhole channels while the classifier is trained with signals over a full-rank channel . . . . 74
4.17 PCC for the modified DDrCNN over keyhole channels where the classifier was trained with a mixed dataset . . . . 75
4.18 Comparison of the classification performance of modified DDrCNN trained on different datasets . . . . 75
4.19 Confusion matrices showing the classification performance of the modified DDrCNN on MIMO keyhole signals at different SNR levels, using models trained on separate datasets . . . . 77
4.20 Comparison of AMC performance of modified DDrCNN-based classifier with the baseline HOC-based ANN AMC over MIMO keyhole channel for 3-modulation pool. . . . . 78

TH-4097_156102033

TH-4097_156102033

# List of Tables

2.1 Layers of DrCNN . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18

2.2 Layers of DDrCNN . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21

2.3 Details of the simulated communication channel for dataset generation . . . . . 23

2.4 Layers of different CNNs considered in ablation study . . . . . . . . . . . . . 25

2.5 Layers of DBCNN . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 29

2.6 Comparison of number of trainable parameters . . . . . . . . . . . . . . . . . 35

2.7 Class-wise and overall percentage classification accuracy (PCC) of DrCNN, DBCNN and DDrCNN at SNR = 4 dB. . . . . . . . . . . . . . . . . . . . . . 37

3.1 Notations used in the cooperative system model and in Algorithms 3.1 and 3.2. 45

3.2 Layers of DDrCNN for input signal size 2x64 . . . . . . . . . . . . . . . . . 50

3.3 PCC comparison between single node and averaging-based fusion with 10 nodes for 64 samples input . . . . . . . . . . . . . . . . . . . . . . . . . . . . 51

4.1 Architectures of Maxpool CNN and 1D CNN . . . . . . . . . . . . . . . . . 62

4.2 Comparison of the number of trainable parameters and training time for the networks under consideration. . . . . . . . . . . . . . . . . . . . . . . . . . 68

TH-4097_156102033

TH-4097_156102033

# 1

# Introduction

TH-4097_156102033

Automatic modulation classification (AMC) is an integral part of cognitive-communication receivers. Traditionally, AMC has been a key technology for military communication [1] applications such as electronic warfare, intruder signal detection, and surveillance. With the advent of 5G and beyond, communication systems need to handle numerous devices in the network to process various information sources. Integrating a large number of devices into the communication networks demands increased cognition in the system. Hence, the realisation of an intelligent communication system that will automatically adjust the parameters to adapt as per the requirements of the situation is much needed. In these types of cognitive radio applications, a known pool of modulations can be used at the transmitter, and a particular modulation is dynamically selected according to the data rate and channel conditions. This is typically achieved by adding overhead pilot symbols at each frame of the signal. However, as the number of digital devices added to the network is increasing exponentially, adding overhead symbols affects the efficiency of the system.

AMC-based intelligent receivers may be employed in these cases to alleviate the problem. Further, the identification algorithm shall provide a high degree of flexibility in sensing different signals under various conditions. For example, signals may be transmitted with single or multiple antennas, and they may employ diverse modulation schemes and may be affected by multi-path fading. Also, the algorithm should ideally require minimal preprocessing, such as filtering, synchronisation, or feature extraction, thus reducing latency and processing overhead because the AMC systems are incorporated in real-time and resource-constrained conditions like embedded systems and edge devices. We shall now look into the mathematical representation of the AMC problem.

## 1.1 General Definition of AMC Problem

In most of the AMC methods, the received base-band signal can be modelled as,

$$y(t) = Ae^{j(2\pi\Delta f_c t+\phi)} \sum_{k} s_k^i p(t - kT_0) + w(t) \tag{1.1}$$

where $A$ is the signal amplitude, $\Delta f_c$ is the frequency offset, $\phi$ is the phase offset, $s_k^i$ is the symbol transmitted from a set $\Omega^i$ containing symbols corresponding to different modulation schemes, $p(t)$ is the pulse shape used with the corresponding period $T_0$ and $w(t)$ is the white Gaussian noise. In general, the modulation classification (MC) problem is to determine the modulation scheme of $s_k$ from the received signal $r(t)$.

AMC is a widely researched area. The initial works in AMC mainly focused on Single Input, Single Output (SISO) systems. With the emergence of multiple-input and multiple-output (MIMO) systems, the AMC methods have been extended to MIMO systems. In the earlier literature, the problem of AMC was mainly tackled using two methods, namely the Likelihood-based (LB) and the Feature-based (FB) methods. With the widespread popularity of Deep Learning (DL), researchers started utilising DL-based classifiers for modulation classification. These three methods are outlined below.

## 1.2 LB Method for AMC

In this method, the likelihood function of the received signal is evaluated under the hypothesis that various signals are received, and classification is done based on the maximisation of the same [2–4]. To evaluate the likelihood function, we need to estimate different signal parameters like phase, frequency, timing parameters, etc. The problem is modelled as a composite hypothesis testing problem. Suppose we have a communication transmitter that selects a modulation type from a set of possible modulations, denoted as $\mathcal{M}$ and let $H_M$ denote the hypothesis that the transmitted message is modulated using a specific modulation type $M$ (where $M \in \mathcal{M}$). Under this hypothesis, the likelihood function of the received signal is represented by $f_M(Y)$, where $Y$ is the received signal.

To determine the modulation type of the received signal, we evaluate the likelihood function for all possible modulation types in the set $\mathcal{M}$. Specifically, we calculate $f_M(Y)$ for each modulation type $M$ in $\mathcal{M}$. The modulation candidate responsible for producing the largest value of the likelihood function is then selected as the modulation type for the received signal. In other words, the modulation type $M$ that maximises the likelihood function $f_M(Y)$ is chosen as the estimated modulation type for the received signal. To compute the likelihood function, it is nec-

TH-4097_156102033

essary to estimate several unknown signal parameters such as carrier phase, frequency offset, symbol timing, and noise variance. Consequently, the problem is formulated as a composite hypothesis testing problem.

Under the additive white Gaussian noise (AWGN) assumption, the received base-band signal can be expressed as

$$Y = s_M(\boldsymbol{\theta}) + n, \tag{1.2}$$

where $s_M(\boldsymbol{\theta})$ denotes the noiseless signal corresponding to modulation $M$ with a set of unknown parameters $\boldsymbol{\theta}$, and $n$ is the additive white Gaussian noise vector with independent and identically distributed samples $n_i \sim \mathcal{CN}(0, N_0)$.

Given the Gaussian and independent nature of the noise, the conditional probability density (likelihood function) of observing $Y$ under hypothesis $H_M$ is given by

$$f_M(Y) = \frac{1}{(\pi N_0)^N} \exp\left( -\frac{1}{N_0} \sum_{k=1}^{N} \left| y_k - s_{M,k}(\boldsymbol{\theta}) \right|^2 \right), \tag{1.3}$$

where $N$ is the number of received samples.

To determine the modulation type of the received signal, $f_M(Y)$ is evaluated for each possible modulation type $M$ in the candidate set $\mathcal{M}$. The modulation type corresponding to the maximum likelihood value is selected as the estimated modulation of the received signal, which can be expressed as

$$\hat{M} = \arg\max_{M \in \mathcal{M}} f_M(Y), \tag{1.4}$$

or equivalently, by minimising the Euclidean distance between the received signal and the hypothesised transmitted waveform as

$$\hat{M} = \arg\min_{M \in \mathcal{M}} \sum_{k=1}^{N} \left| y_k - s_{M,k}(\hat{\boldsymbol{\theta}}_M) \right|^2 . \tag{1.5}$$

In summary, the process involves:

- Evaluating the likelihood function $f_M(Y)$ for each modulation type $M$ in the set $\mathcal{M}$.
- Selecting the modulation type $M$ that maximises the likelihood function as the estimated modulation type of the received signal.

TH-4097_156102033

Based on how the classifier deals with the unknown parameters, the LB classifiers are further divided into ALRT(Average likelihood ratio test), GLRT(Generalised likelihood ratio test) and HLRT(Hybrid likelihood ratio test) [1,3,4] classifiers. Although the likelihood-based approach is optimum in maximising the probability of correct detection, it demands a huge computational power. Since signal detection calls for real-time processing, the method needs to be less complex.

## 1.3 FB Method for AMC

The FB methods extract features from the signals, and the feature vector serves as the tool for classification [5–7]. Examples of features include the statistics of instantaneous amplitudes, phase and frequency, the cumulants and the cyclostationary features. Although FB methods are sub-optimal, they reduce the computational complexity of the classification system over the LB methods. In the FB method, researchers mainly concentrated on higher-order cumulant (HOC) based classifiers and cyclic cumulant-based classifiers.

### 1.3.1 HOC based AMC

The cumulants of a random variable $X$ are derived from its moments. They summarise the shape of a probability distribution. The cumulant generating function (CGF) $K_X(t)$ is defined as the natural logarithm of the moment generating function (MGF):

$$K_X(t) = \log M_X(t)$$

where, $M_X(t) = \mathbb{E}(e^{tX})$.

The cumulants are the coefficients of the Taylor series expansion of $K_X(t)$ about $t = 0$ and obtained by taking derivatives of the CGF with respect to $t$ at $t = 0$. Specifically, the $n$-th cumulant $\kappa_n$ is given by:

$$\kappa_n = \left. \frac{d^n K_X(t)}{dt^n} \right|_{t=0}$$

The cumulants of a random variable are functions of its moments, but they have some advantages over moments. The cumulant value for a Gaussian random variable is zero for orders

TH-4097_156102033

greater than 2. Thus, the HOCs are resistant to Gaussian noise. This property is exploited when cumulants are used as classification features.

In [8], a detailed study of the application of the cumulants on the AMC is presented. Mathematical equations for higher-order cumulants were discussed, and the theoretical values of these cumulants for various modulation schemes were derived. Based on these values, a hierarchical scheme was proposed for the classification of these modulation schemes. Notable works investigating HOC-based AMC in the literature can be found in [9–14].

### 1.3.2 Cyclic Cumulants based classifiers

A cyclostationary process is a signal having statistical properties that vary cyclically with time. Cyclic cumulants are the amplitudes of the Fourier-series components of the time-varying cumulant function for a cyclostationary signal [15]. They represent the periodicity correlation in the statistics of such a process. When the signal is stationary, the cyclic cumulants become conventional cumulants. A number of works are available in the literature where the cyclic cumulants were employed to classify the modulation schemes in different channel environments [16–21]

The drawback of the above FB methods is that the selection of the features for a particular classification problem is manual. The set of best features may vary depending on the problem under consideration. The estimation of these features is complex and computationally intensive.

## 1.4 Deep Learning for AMC

The developments in deep learning (DL) algorithms for computer vision applications opened the door to eliminating the need to extract classification features. The possibility of applying DL in AMC is investigated in [22]. The authors analysed the effectiveness of the Convolutional Neural Network(CNN) based classifier for AMC. They proposed representing the complex base-band communication signal as a two-dimensional image by separating its real and imaginary parts. Studies constituting the application of various DL networks for AMC were reported, and the results enforced the superiority of the DL methods over LB and FB classifiers.

TH-4097_156102033

## 1.5 Existing Works on DL-based AMC

DL-based classifiers were first investigated on SISO signals in [22]. In this work, the authors considered a modulation pool consisting of 11 modulation classes, namely, BPSK, QPSK, 8PSK, 16QAM, 64QAM, BFSK, CPFSK, PAM4, WB-FM, AM-SSB, and AM-DSB. The dataset for the study was generated in GNU Radio using the radio channel model proposed in [23]. Two CNNs with two convolutional layers and two dense layers with different number of neurons in each layer were trained using this dataset. The results of these classifiers were compared with the Deep Neural Network(DNN), the K Nearest Neighbour (KNN) and the support vector machine (SVM) classifiers trained on expert features. The DL networks employed for classification were considerably larger for applications like AMC, and there was scope for improvement in the classification performance.

In [24], the application of DL toward the physical layer has been discussed. The authors proposed a more compact CNN for the modulation classification task. However, the results show that the network failed to distinguish between the 16QAM and 64QAM in the data set. In [25], a classifier consisting of two CNNs was proposed to eliminate the problem of QAM detections. The first CNN called Dropout CNN (DrCNN) was used to distinguish the modulation schemes other than QAMs. The authors employed a constellation image-based dataset to classify the QAMs.

A data-driven model for automatic modulation classification based on Long Short Term Memory (LSTM) is proposed in [26]. It is demonstrated that the LSTM network data representation in the form of magnitude and phase offers better classification accuracy. The authors have reported that CNN classifiers outperform LSTM classifiers at lower SNRs.

A comprehensive analysis of the application of DL considering 24 modulation schemes widely used in today's communication systems is elaborated in [27]. ResNet and VGG net architecture-based classifiers were employed to classify these modulation classes. These classifiers were designed for SISO signals and outperformed their FB counterparts.

A cooperative cumulants-based classifier for distributed communication networks was proposed in [28]. The authors proposed a fusion centre (FC) based classification architecture where

TH-4097_156102033

the individual nodes transmit the HOC features to the FC, and a maximum likelihood (ML) combining is utilised at the FC for the final decision. The work mainly concentrated on classifying QPSK and 16QAM modulation schemes in a Rayleigh fading environment. The selection of HOC features and the ML combining tend to be complex when the number of classes increases.

In the DL-based AMC method proposed in [29], Wang et al. presented an enhanced approach designed explicitly for MIMO signals. To address the channel's influence, they incorporate a zero-forcing (ZF) equaliser within the system. While this ZF equaliser effectively mitigates channel effects, it introduces additional complexity to the overall design, as the DL network needs to be trained on the equaliser's output. This increases the complexity of the classifier.

A cooperative AMC method for MIMO signal classification based on DL was proposed in [30]. A one-dimensional CNN-based classifier was trained on the signals from each of the receiving MIMO antennas. A cooperative mechanism was then employed to choose the modulation from the CNN predictions for each of the MIMO receiving antennas. A $2 \times 4$ MIMO Rayleigh channel was considered for this study. They were able to outperform FB-based methods for a modulation pool consisting of {BPSK, QPSK, 8PSK, 16QAM}. However, the network they employed was much larger than the previous classifiers.

It is reported in [31], that conventional cumulant-based classifiers failed to classify the modulation classes under rank-deficient channels. The authors propose a modified cumulant-based classifier for lower-order PSKs for this channel. The method failed to classify higher-order modulation schemes.

## 1.6 AMC using shallow CNNs

Deep Learning models, particularly CNNs, learn features directly from the raw data (or simple pre-processed data), eliminating the need for hand-crafted feature engineering. They are highly robust to channel impairments like noise, fading, and frequency offsets. The above literature review indicates that researchers have explored both state-of-the-art deep CNN architectures such as ResNet and VGGNet, as well as lightweight models like DrCNN, for AMC tasks. Although deep CNN architectures such as ResNet-50 possess remarkable representa-

TH-4097_156102033

tional capacity, they present significant limitations when used for AMC. Their high computational complexity makes them unsuitable for real-time, low-power, or edge-device applications. In addition, they require extensive amounts of labeled data for effective training, which is often impractical in AMC scenarios. Deep networks are also prone to overfitting, as they may easily memorize noise from the training data unless carefully regularized.

In contrast, shallow CNNs (SCNNs) offer a more balanced trade-off between efficiency and accuracy [24]. An SCNN in this context typically consists of two to four convolutional layers followed by one or two fully connected layers. Such architectures are lightweight, faster to train, and more adaptable to limited datasets, making them particularly effective for resource-constrained AMC tasks. SCNNs tend to fit AMC problems well, mainly because radio signals themselves are much simpler than natural images. When working with images, deep networks need several layers to recognize basic edges, then textures, and finally complex objects. In contrast, the differences between modulation types—say, between QPSK and 16QAM—can often be picked up by just a few well-designed convolutional filters. These filters are sensitive to the signal's phase shifts, amplitude variations, and constellation shapes, which are usually enough to distinguish one modulation from another. Another big advantage is computational efficiency. With fewer layers and parameters, SCNNs are lighter, faster, and easier to deploy on low-power devices such as drones or field radios. They train quickly and run smoothly even on modest hardware, which makes them perfect for real-time or edge applications. They also handle limited data better. In many RF setups, collecting huge labeled datasets is expensive and time-consuming. Because shallow networks are less prone to overfitting, they can still learn useful patterns from smaller datasets and perform reliably in the field [26].

Considering these factors, this thesis adopts a shallow CNN architecture for AMC, aiming to achieve a balanced trade-off between computational efficiency, accuracy, and robustness under limited-data and resource-constrained conditions.

## 1.7 Datasets used in DL-AMC

Most of the works in DL-AMC employed RadioML datasets for classifier evaluation. The RadioML 2016.04C is a synthetic dataset [22], generated with GNU Radio, consisting of 11

TH-4097_156102033

modulations. This is a variable-SNR dataset with moderate frequency offset drift, light fading, and numerous different labelled SNR increments for use in measuring performance across different signal and noise power scenarios. The latter dataset from the same team, RadioML 2018.01A, includes synthetic simulated channel effects of 24 digital and analogue modulation types, which have been validated with over-the-air signals [27]. Data are stored in HDF5 format as complex floating point values, with 2 million examples, each 1024 samples long. Another dataset available is HisarMod 2019 [32]. The dataset includes 26 modulation types from 5 different modulation families, which are analog, FSK, PAM, PSK, and QAM. In the dataset, there are 1500 signals, each of which has a length of 1024 I/Q samples, for each modulation type.

Many researchers also generate their own synthetic datasets using tools like MATLAB or GNU Radio, allowing for specific control over modulation types, channel impairments (AWGN, fading, frequency offset, sampling time drift, etc.), and SNR ranges relevant to their own research. Some studies also aim to collect and use real-world over-the-air data, though this is often more challenging to make publicly available due to practical constraints. In this thesis, the majority of results are based on datasets synthetically generated using MATLAB's Deep Learning Toolbox for AMC. Additionally, for the SISO AMC scenario, classifier performance is also evaluated using signals from the RadioML 2018.01A dataset.

## 1.8 Motivation and Problem Formulation

Based on the studies made on the existing literature, the following research gaps are identified.

**1.** The classifier proposed by Wang et al. in [25] is complex due to the various types of data required for training. Their classifier's first CNN (DrCNN) utilised in-phase and quadrature (IQ) data representation for training and testing. DrCNN applied dropouts as the regulariser in each of the CNN layers. The effectiveness of regularisation methods for CNNs when applied to communication problems is poorly understood and requires detailed investigation.

**2.** The rise of distributed communication networks and cognitive radio (CR) networks highlights cooperative classification as an appealing approach to enhance the AMC performance. In

TH-4097_156102033

a SISO system, combining signals from multiple sensors has been shown to yield improved statistical estimation, thereby enhancing AMC performance [28]. However, existing studies on the DL-based cooperative AMC have yet to explore scenarios involving multiple receiving nodes.

**3.** There are only a few studies in the literature that delve into the AMC for MIMO signals. In [29, 30], the authors utilised a one-dimensional CNN for the classification. They employed a Rayleigh MIMO channel for their experiments. It is worth exploring the potential for enhancing both the classification accuracy and the performance of DL-based classifiers for MIMO signals across various channel models. Further HOC-based AMC systems have struggled to classify higher-order PSK modulation schemes in scenarios where the MIMO channel is rank-deficient. Existing research on DL-AMC has ignored the challenges posed by rank-deficient channels, such as keyhole channels. Keyhole channels can arise in environments where signals propagate through narrow openings or windows in large buildings, leading to limited spatial diversity despite multiple antennas. This makes it important to study DL-AMC in keyhole channel conditions and assess its performance.

The thesis addresses the above problems using shallow CNNs. The channel models considered in this work—the Rician channel in Chapter 2, the Rayleigh channel in Chapter 3, and the MIMO Rayleigh and keyhole channels in Chapter 4—were chosen to span line-of-sight, rich-scattering non-line-of-sight, and rank-deficient propagation conditions. These are the channel models predominantly adopted in the AMC literature at the time this research commenced, and they provide an analytically tractable and reproducible framework that isolates the effect of small-scale fading on the classification performance, while enabling a fair comparison with the existing methods. A time-invariant, block-fading model is assumed throughout, in which the channel remains constant over the observation window of $N = 128$ samples; this flat-fading assumption is valid when the channel coherence time is much larger than the observation window, as in the low-to-medium mobility scenarios of interest such as pedestrian and indoor environments. It is acknowledged that these idealised models do not capture the full complexity of standardised channel models such as the 3GPP profiles adopted in the industry—which additionally incorporate frequency-selective multipath, delay-spread profiles, Doppler effects, and

TH-4097_156102033

spatial correlation—and the extension of the proposed classifiers to such 3GPP-compliant channels is identified as a direction for future work in Chapter 5.

### 1.8.1 Thesis organisation

The rest of the thesis is organised as follows:

**Chapter 2** comprises the proposed SCNN classifiers for eight digital modulation schemes: BPSK, QPSK, 8PSK, GFSK, CPFSK, PAM4, 16QAM, and 64QAM for a SISO communication channel. Two SCNN architectures for performance improvement are considered, and their classification performance is studied and compared with the existing classifiers in the literature.

**Chapter 3** evaluates how cooperating between different receiver nodes improves the classification accuracy of CNN-based AMC for a SISO channel. The classification performance for different methods of cooperation is evaluated for the same set of eight modulation schemes.

In **Chapter 4**, the CNN-based AMC is extended to the MIMO channels, considering a pool of modulation schemes: BPSK, QPSK, OQPSK, 8-PSK and 16-QAM. The classifier's performance is evaluated with competing classifiers on the MIMO dataset. At the MIMO receiver, cooperation is applied to improve the classification accuracy. The suitability of the proposed CNN-based classifier is evaluated on keyhole signals, and its performance was compared with that of an FB-based method proposed in the literature for keyhole channels. **Chapter 5** features the conclusions on the work done in the thesis. A few research directions for the future research are also identified.

TH-4097_156102033

# 2

# AMC of SISO signals using Shallow CNNs

TH-4097_156102033

## 2.1 Introduction

Traditionally the AMC problem was based on feature extraction and further classification based on these features. The emergence of DL methods has helped develop a data-driven approach rather than an FB method. The previous chapter introduced the AMC for SISO signals based on DL. This chapter explores the existing SCNN-based classifiers for AMC and examines the scope of improving their classification performance. The aim of the study is to keep the classifier compact so that it can be included in resource-constrained AMC applications.

An SCNN architecture comprising two convolutional layers and a 256-neuron fully connected layer was proposed in [22] for the AMC of eight digital and three analogue modulation types, namely BPSK, QPSK, 8PSK, GFSK, CPFSK, PAM4, 16QAM, 64QAM, AM-DSB, AM-SSB and WB-FM. This CNN was compared with Deep Neural Networks (DNN) based classifiers and a Support Vector Machine (SVM) classifier, trained on expert features, demonstrating superior classification performance. The CNN incorporated 2.5 million trainable parameters, which is considerably high considering the demands of communication systems in which the AMC system is employed. In an effort to reduce the complexity of this CNN, O'Shea and Hoydis, in their subsequent work [24], proposed a new CNN having a lower number of trainable parameters. However, the classification performance for QAMs was poor. An analysis of the classification results showed that the network failed to distinguish between 16QAM and 64QAM.

Wang et al. [25] solved the above problem by using a combination of two CNNs as shown in Fig.2.1, at the cost of increased complexity. This study focuses on a modulation class encompassing eight modulation schemes: The first CNN used dropouts to combat over-fitting and was called DrCNN. Dropouts are placed after each of the convolution layers and the dense layers. The second CNN was used to differentiate between 16QAM and 64QAM. The authors trained DrCNN using IQ data and the second CNN using the constellation image data synthesised from the received signal. DrCNN is able to detect QAMs as a single class for SNRs as low as $-6$ dB. If DrCNN outputs the QAM class, the constellation image of the input signal is presented to the second CNN. These data are applied to the second CNN through a density window, which

TH-4097_156102033

divides the density of constellation points into three degrees: higher density, moderate density, and lower density, and yellow, green, and blue colours are applied to represent these three degrees, respectively. Considering both the CNNs, a total of seven convolutional layers and seven fully connected layers are included in their classifier. The performance of DrCNN is compared with that of the RNN, the DNN, and the Inception network-based classifiers. The results show that DrCNN outperforms its peers when the QAMs are clubbed together as a single class. While DrCNN demonstrated superior classification accuracy compared to others, the number of trainable parameters was on the higher side. The architectural details of DrCNN are outlined in Table 2.1. Notably, dropout layers are incorporated after each convolutional and dense layer. However, the inclusion of dropouts in convolutional layers is a topic of widespread debate in the literature. This research critically examines this issue and introduces a classifier capable of effectively categorising all eight modulation schemes, taking into consideration the following factors:

- To reduce the complexity of the classifier by limiting it to a single CNN, unlike the existing DrCNN-based classifier.
- To boost the classification performance while maintaining a lightweight CNN.

In the DrCNN architecture, dropouts are included to mitigate the overfitting problem encountered in the neural networks. In the following Section 2.2, the problem of overfitting in CNNs and the various methods to alleviate the same are discussed. The proposed SCNN based classifier is introduced in Section 2.3. In Section 2.4, the details about the implementation of the classifier and the results of the simulations are included. A summary is presented in Section 2.5.

## 2.2 Strategies to Mitigate Overfitting in CNNs

Overfitting is a serious problem of CNNs affecting their performance [33]. A deep learning model tends to learn not only the underlying patterns but also the statistical noise and the outliers present in the training data. The primary objective in training is to minimise the loss function across all neurons. Over-fitting occurs when a neuron adjusts itself to compensate for

TH-4097_156102033

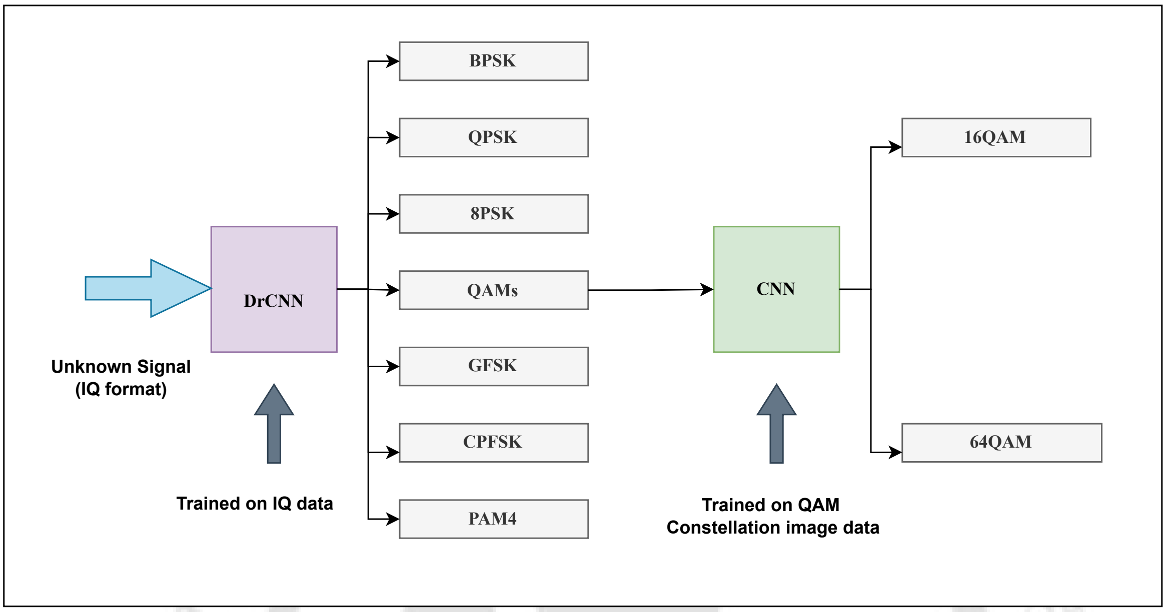



Figure 2.1: A schematic diagram of the existing DrCNN-based classifier for the AMC of eight modulation classes.

errors made by other units, resulting in intricate co-adaptations. Hence, the hidden units in the neural networks will have highly correlated behaviour. These complex co-adaptations need to generalise more effectively to unseen datasets, giving rise to the over-fitting problem. As a result, the trained network will perform poorly on the validation dataset. Regularisation is a set of methods for reducing overfitting in machine learning models. Common regularisation methods that are utilised in DL includes, $L_1$ and $L_2$ regularisation, dropouts, data augmentation, early stopping and batch normalisation (BN). It is reported that dropouts perform better than $L_1$, $L_2$ regularisers [34] and have become very popular in the literature. Wang et al. [25] employed dropouts as the regulariser in their CNN. In the succeeding section, dropouts and batch normalisation techniques are elaborated.

### 2.2.1 Dropouts as a regulariser

Dropout is a regularisation technique in deep learning that became very popular after the work of Srivastava et al. [35], which presented an exhaustive result on the effectiveness of dropouts in dense layers as well as convolutional layers. Experiments on the CIFAR-10 dataset for image processing showed that the test error has been reduced from 15.60% to 12.61% by applying dropouts in convolutional and hidden layers.

TH-4097_156102033

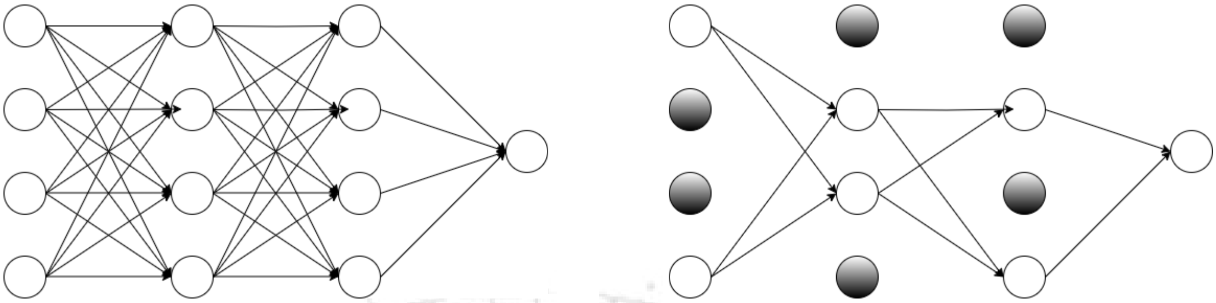



Figure 2.2: Dropout applied to a standard neural network

The term ”dropout” denotes the random exclusion of nodes (both in the input and hidden layers) within a neural network, as depicted in Fig. 2.2. During dropout, all forward and backward connections associated with the omitted node are temporarily eliminated during training, leading to a modified network architecture derived from the original parent network. The dropout probability, denoted as $p$, determines the likelihood of the nodes being excluded. The value of $p$ typically varies between 0.2 and 0.5.

The implementation of dropout serves as a remedy by preventing units from exclusively compensating for others' errors. This is achieved by introducing unreliability into the presence of units during each iteration. By randomly omitting a subset of nodes, dropout compels the layers to assume varying degrees of responsibility for the input, adopting a probabilistic approach that mitigates the risk of over-fitting.

### 2.2.2 Batch Normalisation (BN)

BN was introduced in [36] to speed up the training process of ML algorithms by removing the covariance shift experienced by the different layers of the network. Covariate shift can be defined as the change in the distribution of network activations due to the change in network parameters during training. BN is implemented in CNNs as another layer, which receives the output from one layer and normalises it before passing it on as input to the next layer. The output, $Y$, of the BN layer is given by,

$$Y = \gamma \cdot \bar{X} + \beta \tag{2.1}$$

TH-4097_156102033

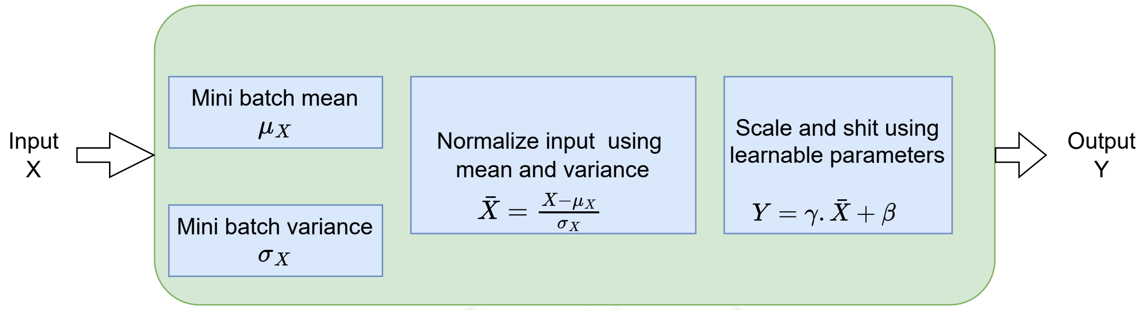



Figure 2.3: The process block diagram of a BN layer.

and,

$$\bar{X} = \frac{X - \mu_X}{\sigma_X} \tag{2.2}$$

where $\mu_X$ , $\sigma_X$ are the mean and standard deviation of input to the BN layer computed over a mini batch of size less than that of the training dataset. BN has two learnable ($\gamma$, $\beta$) as well as two non-learnable parameters ($\mu_X$,$\sigma_X$) associated with it . The introduction of BN improves the generalisation of the network and thereby reduces overfitting. The process block diagram of a BN layer is shown in Fig.2.3.

In the following section, two CNNs based on dropouts and BN are proposed, and their AMC performances are assessed.

Table 2.1: Layers of DrCNN

| **Layer** | **Output dimensions** |
|---|---|
| Input | (2,128) |
| Convolution2D (Filters 128, (2,8)) + PReLU | (121 ×128) |
| Dropout (0.5) | / |
| Convolution2D (Filters 64, (1,16))+ PReLU | (114×64) |
| Dropout (0.5) | / |
| Flatten | 7296 |
| Dense + PReLU | 128 |
| Dropout (0.5) | / |
| Dense + PReLU | 64 |
| Dropout (0.5) | / |
| Dense + PReLU | 32 |
| Dropout (0.5) | / |
| Dense + Softmax | Number of classes |

TH-4097_156102033

## 2.3 Proposed SCNN Classifiers for the AMC of 8 Digital Modulation Classes.

The structure of the proposed SCNN variants differs according to the inclusion of dropouts and BN after the dense layers. Accordingly, the SCNNs are named as the ***dense layer dropout*** CNN (**DDrCNN**) and ***dense layer batch normalisation*** CNN (**DBCNN**) respectively.

### 2.3.1 DDrCNN for AMC

In the previous section, it was explained how dropouts are effective in fighting the overfitting problem. However, various studies [37,38] have reported that dropouts in the convolutional layer lead to diminished performance of CNN classifiers. By removing dropouts after the convolutional layers, the feature set may improve. Hence, the dense layers may be able to perform better without the requirement of a second CNN to classify between the QAMs. Such a scheme will significantly simplify the CNN-based classifier for AMC. A dense layer dropout CNN-based classifier for AMC is proposed in the first section, and the effect of Batch Normalisation applied in the layers of CNN on AMC is studied in the second section.

The investigation commenced by examining the impact of dropouts in these layers on classification accuracy. The removal of dropouts from the convolutional layers improved the result of classification. Further, the reduced version of DrCNN with dense layer dropout was able to outperform DrCNN on a pool of 8 modulation schemes. Based on the findings, a DDrCNN-based classifier is proposed as schematically shown in Fig. 2.4.

Table 2.2 shows the architecture of DDrCNN. The convolutional neural network consists of two convolutional layers. The first layer uses 128 filters with a kernel size of (2, 6), followed by a second layer that applies 64 filters with a kernel size of (1, 14). These are succeeded by three fully connected (dense) layers containing 64, 32, and 11 neurons, respectively. The final dense layer employs a softmax activation function to generate the probability distribution across the modulation classes.

A parametric rectified linear unit (PReLU) activation function is employed after each of the convolutional layers and hidden layers, other than the final dense layer. PReLU is a modified version of ReLU where a learnable negative slope parameter is added in the activation function. It is given by (2.3).

TH-4097_156102033

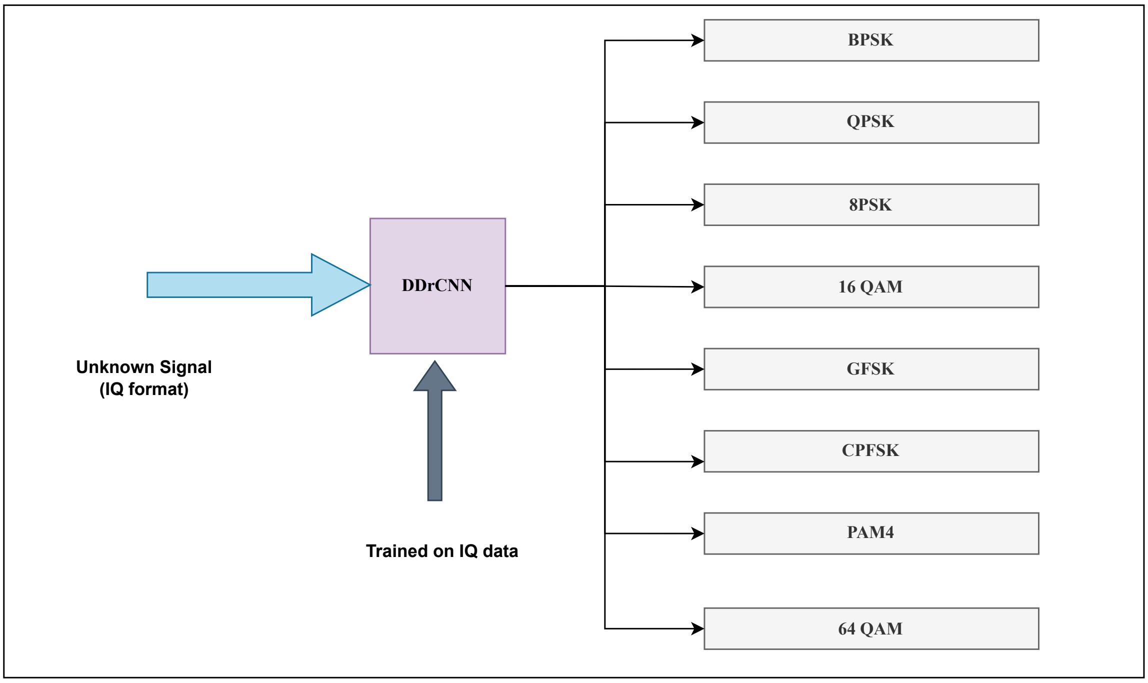



Figure 2.4: A schematic diagram of the DDrCNN-based classifier for the AMC of eight modulation schemes.

$$f(x) = \begin{cases} \alpha \cdot x & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \tag{2.3}$$

where $\alpha$ is a learnable parameter.

The categorical cross-entropy is chosen as the loss function for training DDrCNN. The ground truth of class symbols represented in one-hot encoding ( $\boldsymbol{y}_i$) and the cross entropy of $\hat{\boldsymbol{y}}_i$ is given by,

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{y}_i \cdot log(\hat{\boldsymbol{y}}_i) \tag{2.4}$$

where N is the number of samples considered in one training batch and $\hat{\boldsymbol{y}}_i$ is the predictions of $\boldsymbol{y}_i$.

The idea of dense layer dropout is extended to dense layer normalisation, and a dense layer batch normalisation CNN (DBCNN) is proposed in the next section.

TH-4097_156102033

### 2.3.2 DBCNN for AMC

As described above, BN is a technique to standardise the inputs to a layer for each mini-batch while training deep neural networks. BN works differently during training and testing. During training, the layer normalises its output using the mean and standard deviation of the current batch inputs according to the equations 2.1 and 2.2. Practically $\gamma$, the learnable scaling factor is initialised as one and $\beta$, is initialised as zero. During the prediction, the layer normalises its output using a moving average (mean) and standard deviation of the batches it has seen during training according to the equation 2.1. Experiments described in a later section have shown that BN applied to the output of hidden dense layers performs better than when BN is applied after all the layers. Hence, a CNN which includes BN after the hidden layers is proposed. Only max-pooling layers, along with the activation layers, are applied after the convolutional layers. Normalisation is applied only for the outputs of the dense layers. Detailed structure of DBCNN is given in Table 2.5.

## 2.4 Implementation Details and Results

The generation of the dataset, as well as the details of the implementation platform, is explained in the following subsections. The ablation studies carried out to validate the efficacy

Table 2.2: Layers of DDrCNN

| Layer | Output dimensions |
|---|---|
| Input | (2,128) |
| Convolution2D (Filters 128, (2,6)) | (123, 128) |
| PReLU | (123, 128) |
| MaxPooling2D | (62, 128) |
| Convolution2D (Filters 64, (1,14)) | (49, 64) |
| PReLU | (49, 64) |
| MaxPooling2D | (25, 64) |
| Flatten | 1600 |
| Dense + PReLU | 64 |
| Dropout (0.4) | / |
| Dense + PReLU | 32 |
| Dropout (0.4) | / |
| Dense + Softmax | Number of classes |

TH-4097_156102033

of the proposed DDrCNN and DBCNN are presented in the following section. The percentage of correct classification (PCC) is used as a metric to evaluate the performance of the classifier. Specifically, for a given Signal-to-Noise Ratio (SNR), PCC is computed as the proportion of correctly classified signals to the total number of test signals at that specific SNR. Mathematically, it is expressed as

$$PCC = \frac{\text{Number of Correct Classifications}}{\text{Total Number of Test Signals}} \tag{2.5}$$

The classification performance of these networks is evaluated for different scenarios, and a comparative evaluation of DrCNN and these proposed networks is also presented.

For the training and testing of CNNs, a computing machine with NVIDIA GeForce GTX 1650Ti, offering 4 GB RAM, is utilised. Deep learning modules are implemented using Keras with TensorFlow as the back end. The details of the datasets used for the training and testing of the classifiers are given below.

### 2.4.1 Dataset

The datasets used in deep learning-driven AMC problems typically consist of synthetic data. O'Shea et al. demonstrated the efficacy of training DL networks using synthetically generated data in their study [27]. Similarly, the authors of the DrCNN-based classifier employed synthetic datasets for both training and testing their classifier. Thus, this chapter mainly utilises a dataset generated using MATLAB for training and testing deep learning networks. A few results on the publicly available RadioML dataset were included in this section for comparative analysis. Since the exact modulation classes are not present in this dataset, a modulation pool (BPSK, QPSK, OQPSK,4ASK, GMSK, 8PSK, 16QAM,64QAM) is extracted. In the original dataset, each of the signals is of length 1024, and a total of 24 modulation classes are considered. While extracting a subset of classes, the signal length is limited to 128.

A Rician multi-path channel model is assumed to generate the synthetic dataset. The details of the channel parameters are given in the Table. 2.3. Signals are modulated at a rate of 4 samples per symbol. The data set generated consists of 3000 instances of each modulation scheme and each instance is 128 samples long. The SNR varies from -10 dB to 10 dB at an

Table 2.3: Details of the simulated communication channel for dataset generation

| **Parameters** | **Specifications** |
|---|---|
| Channel model | Rician fading |
| SNR | -10dB to 10dB |
| Sampling Frequency ($F_S$) | 200 KHz |
| Path delays | [0, 1.8, 3.4]/$F_S$ |
| Average path gains | [0,-2,-10] |
| KFactor | 4 |
| Maximum Doppler Shift | 4 |
| Maximum Clock Offset | 5 |
| Center Frequency | 902 MHz |

interval of 2 dB. The dataset is saved in the HDF5 file format for easy access in the Python environment. It occupied 546 MB of memory space.

### 2.4.2 Selection of the CNN architecture for the AMC

The network under consideration was evaluated for various changes in the structure. The results of these ablation studies are discussed here. During these studies, the training parameters were kept constant. All the networks are trained using 70% of instances from each modulation scheme, each consisting of 128 samples. The remaining 30% of signals are reserved for testing to evaluate the classification accuracy. While training, the number of epochs were set as 200, and an early stopping callback with patience 20 was used. A batch size of 256 was used while training all these networks.

The following experiments were carried out while finalising the proposed classifiers.

**Exp. No 1: Effect of dropouts only on the dense layers:-** The dropouts in the convolutional layers are omitted from the DrCNN, and the network is trained and tested on the synthetic dataset. The result, along with the PCC of DrCNN, is plotted in Fig. 2.5a. The improvement in PCC is evident from the figure. An improvement of around 30% in the PCC is observed for any given SNR. The same networks were trained and tested on the RadioML dataset. The result for the same is plotted in Fig. 2.5b. For this particular dataset, the improvement in performance is smaller than that of the synthetic dataset. This may be due to the fact that the RadioML dataset includes signals modelled using different channel impairments, and the modulation classes considered were different.

TH-4097_156102033

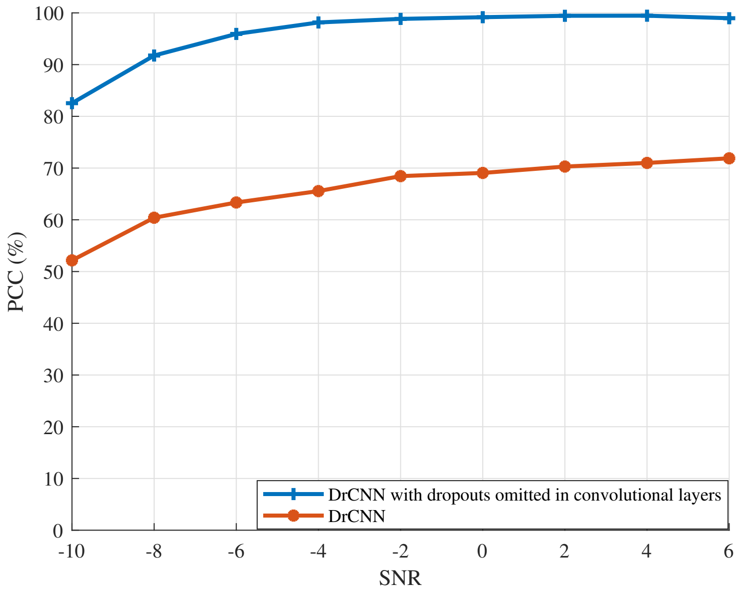
**Exp. No 2: Effect of reducing the number of hidden layers:-** Further studies were conducted to improve the performance as well as reduce the complexity of the network. The two convolutional layers employed in the DrCNN incorporated the kernels of size (2,8) and (1,16), respectively. The network is trained and tested with varying kernel sizes, and observing the slight differences in the PCC, the two convolutional kernels as $\{(2,6),(1,14)\}$ were selected and name the network as CNN1. One of the hidden layers is omitted to trim the network size. The resultant DDrCNN network is trained and tested on the dataset. The training performance of DDrCNN is plotted in Fig. 2.6, which asserts the effect of dropouts on the validation loss. The CNN1, featuring three dense layers, demonstrates superior performance at lower SNRs. However, from 0 dB onwards, DDrCNN underperforms CNN1 by only 1% as shown in the PCC graph Fig. 2.7. Compared to CNN1, which has three dense layers, DDrCNN only has a considerably lower number of trainable parameters. The confusion matrix for the DDrCNN-based classifier for various SNRs is given in Fig. 2.8.

TH-4097_156102033

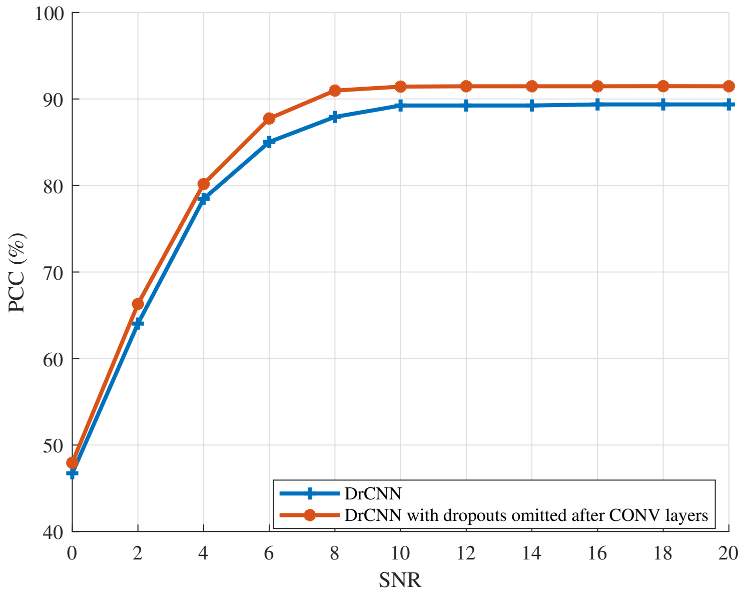
Table 2.4: Layers of different CNNs considered in ablation study

| CNN1 | CNN2 | CNN3 | CNN4 | CNN5 | CNN6 | DDrCNN | DBCNN |
|---|---|---|---|---|---|---|---|
| Input | Input | Input | Input | Input | Input | Input | Input |
| Conv2D | Conv2D | Conv2D | Conv2D | Conv2D | Conv2D | Conv2D+Maxpool | Conv2D+Maxpool |
| PReLU | BN+PReLU | PReLU | BN+PReLU | PReLU | PReLU | PReLU | PReLU |
| Conv2D | Conv2D | Conv2D | Conv2D | Conv2D | Conv2D | Conv2D+Maxpool | Conv2D+Maxpool |
| PReLU | BN+PReLU | PReLU | BN+PReLU | PReLU | PReLU | PReLU | PReLU |
| Flatten | Flatten | Flatten | Flatten | Flatten | Flatten | Flatten | Flatten |
| Dense(128) | Dense(128) | Dense(128) | Dense(128) | Dense(128) | Dense(64) | Dense(64) | Dense(64) |
| PReLU | PReLU | PReLU | BN+PReLU | BN+PReLU | PReLU | PReLU | PReLU |
| Dropout | BN | BN | Dropout | Dropout | BN | Dropout | BN |
| Dense(64) | Dense(64) | Dense(64) | Dense(64) | Dense(64) | Dense(32) | Dense(32) | Dense(32) |
| PReLU | PReLU | PReLU | BN+PReLU | BN+PReLU | PReLU | PReLU | PReLU |
| Dropout | BN | BN | Dropout | Dropout | BN | Dropout | BN |
| Dense(32) | Dense(32) | Dense(32) | Dense(32) | Dense(32) | O/P Dense | O/P Dense | O/P Dense |
| PReLU | PReLU | PReLU | BN+PReLU | BN+PReLU | Softmax | Softmax | Softmax |
| Dropout | BN | BN | Dropout | Dropout | | | |
| O/P Dense | O/P Dense | O/P Dense | O/P Dense | O/P Dense | | | |
| Softmax | Softmax | Softmax | Softmax | Softmax | | | |

TH-4097_156102033




(a) Performance on Synthetic data-set.




(b) Performance on RadioML dataset.

Figure 2.5: Effect of dropouts only in dense layer.

**Exp. No 3: Effect of batch normalisation only in dense layer:-** The effect of normalisation on layers of the CNN is evaluated by employing the Batch Normalisation layer in the architecture. The various CNNs analysed in this ablation study are detailed in Table 2.4. CNN1 contain three dense layers and dropouts only in dense layers, while CNN2 includes BN after the two

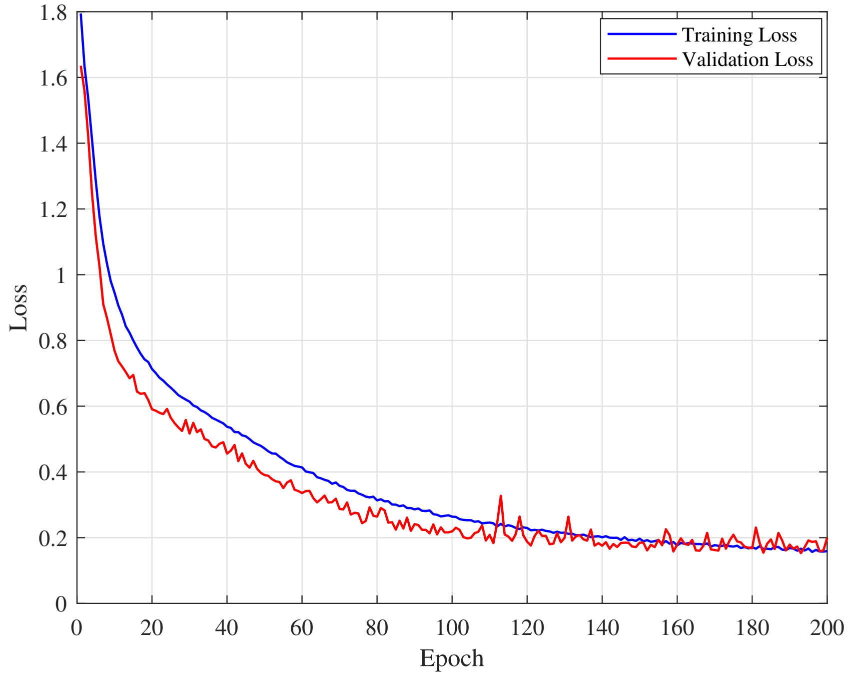



Figure 2.6: Training and validation loss over epochs (DDrCNN).

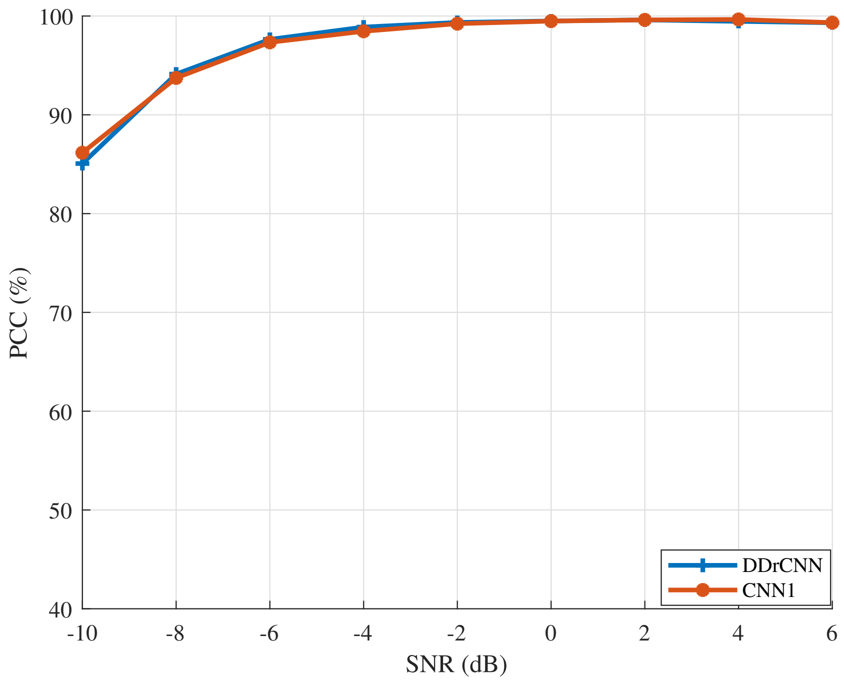
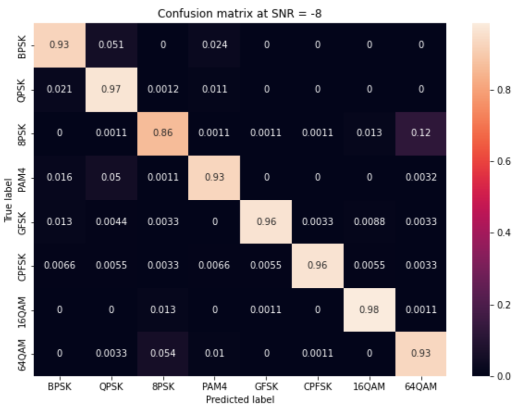
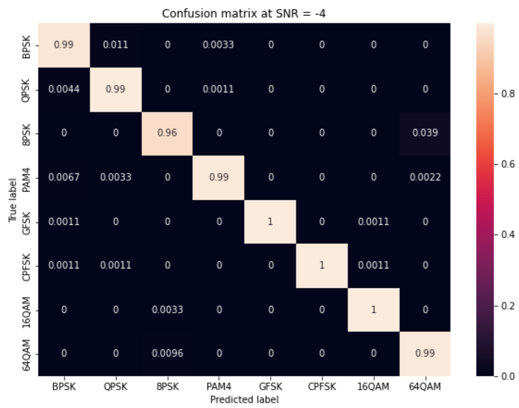



Figure 2.7: Performance of DDrCNN classifier for various SNRs compared with CNN1.

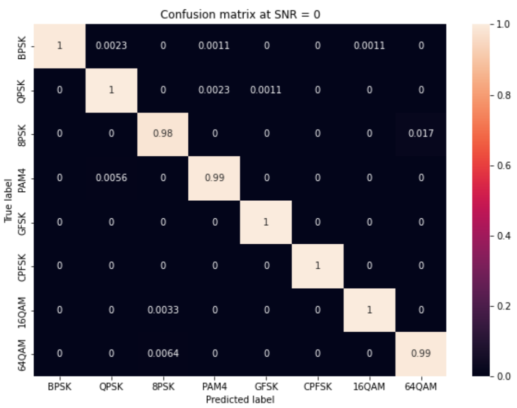
TH-4097_156102033



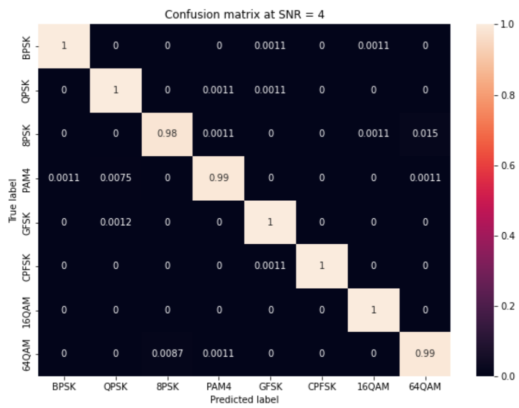

(a) SNR -8 dB




(b) SNR -4 dB




(c) SNR 0 dB




(d) SNR 4dB

Figure 2.8: Confusion matrix of DDrCNN for various SNRs.

TH-4097_156102033

convolutional layers and the three dense layers. CNN3 has BN only after the three dense layers. In CNN4, BN is applied after the convolutional layers and the dense layers, but dropouts are applied only after the dense layers. CNN5 has BN and dropouts after the dense layers only. The PCC for CNNs employing dropouts is compared to that of CNN, which incorporates BN layers. Fig. 2.9 shows that applying BN only in the dense layers returns the best performance among others. Similar to the DDrCNN, max-pooling layers are added to the convolutional layers, and the corresponding classification performance is evaluated. The classification performance of the resultant network (DBCNN) is plotted in Fig. 2.10 along with that of CNN3 and CNN6, which contain only BN as the normalisation layer. The training and validation loss curves are plotted in Fig. 2.11. CNN6 is a trimmed version of CNN3 where one of the hidden layers is omitted. Compared to CNN3, DBCNN has only 20% trainable parameters, but it achieved almost similar classification performance at around 0dB.To evaluate the performance of DDrCNN and DBCNN on the RadioML dataset, the PCC for both networks is plotted in Fig. 2.12. On the RadioML dataset, both networks yielded almost similar classification performance.

Table 2.5: Layers of DBCNN

| **Layer** | **Output dimensions** |
|---|---|
| Input | (2,128) |
| Convolution2D (Filters 128, (2,6)) | ( 123, 128) |
| MaxPooling2D | (62, 128) |
| PReLU | (61, 128) |
| Convolution2D (Filters 64, (1,14)) | (49, 64) |
| Maxpooling2D | (25, 64) |
| PReLU | (24, 64) |
| Flatten | 1600 |
| Dense + PReLU | 64 |
| BN | NA |
| Dense + PReLU | 32 |
| BN | NA |
| Dense + Softmax | Number of classes |

TH-4097_156102033

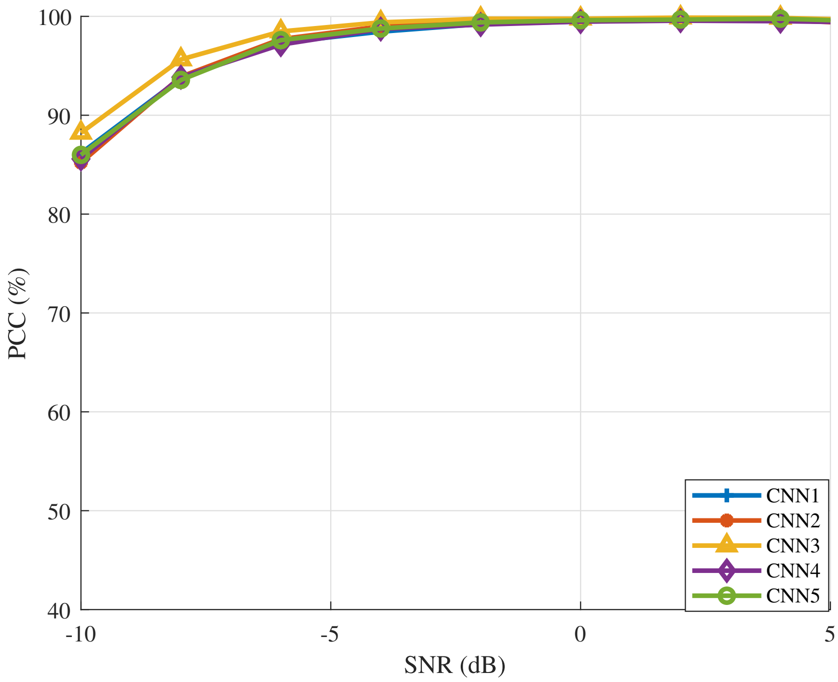



Figure 2.9: PCC plot for various intermediate CNNs listed in Table 2.4.

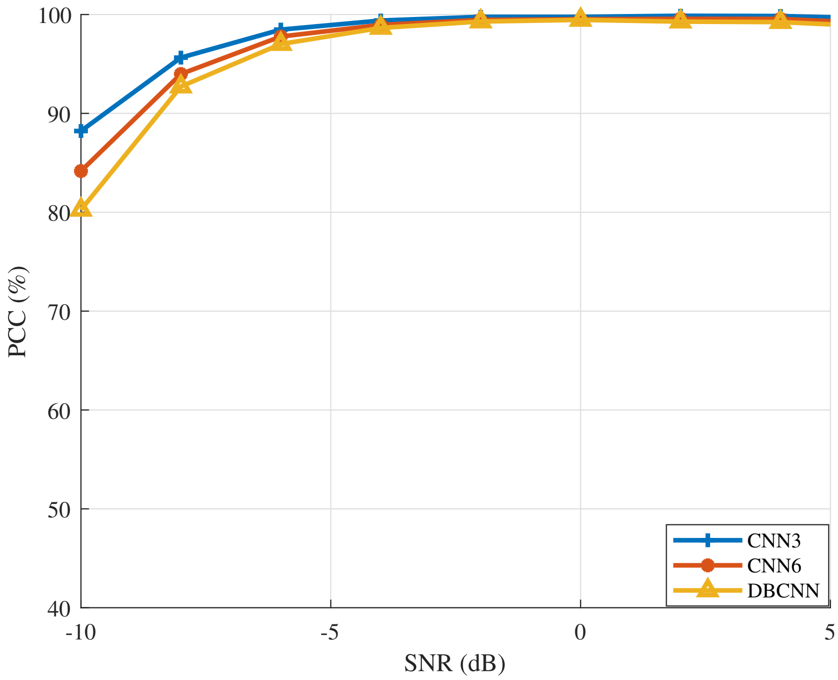
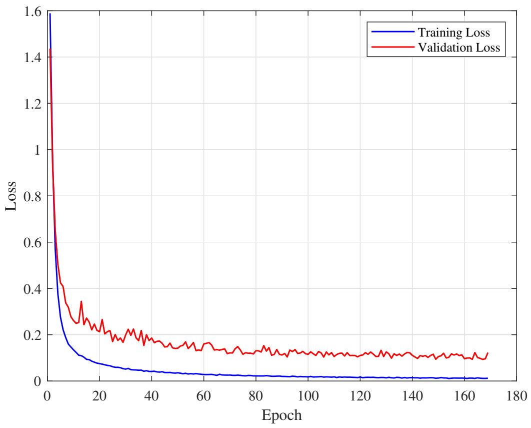
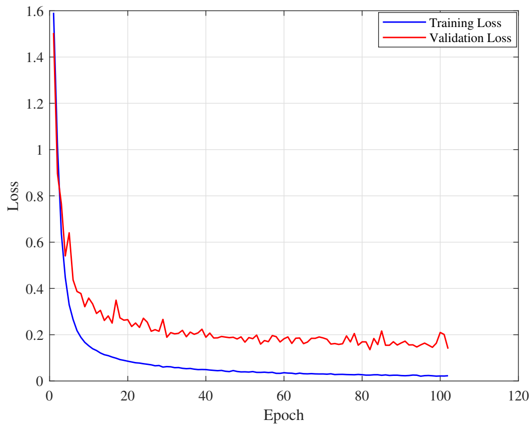



Figure 2.10: PCC of DBCNN compared with CNN3 and CNN6.

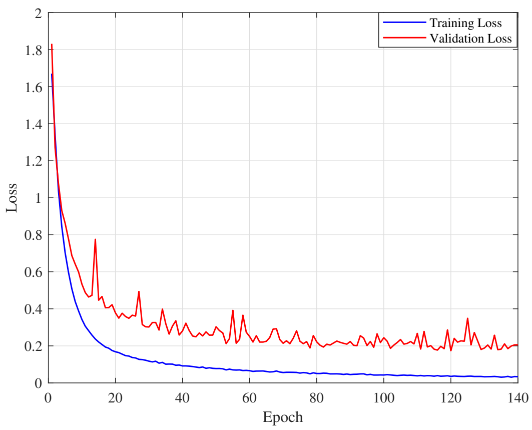
TH-4097_156102033




(a) CNN3




(b) CNN6




(c) DBCNN

Figure 2.11: Training and validation loss for various CNNs .

TH-4097_156102033

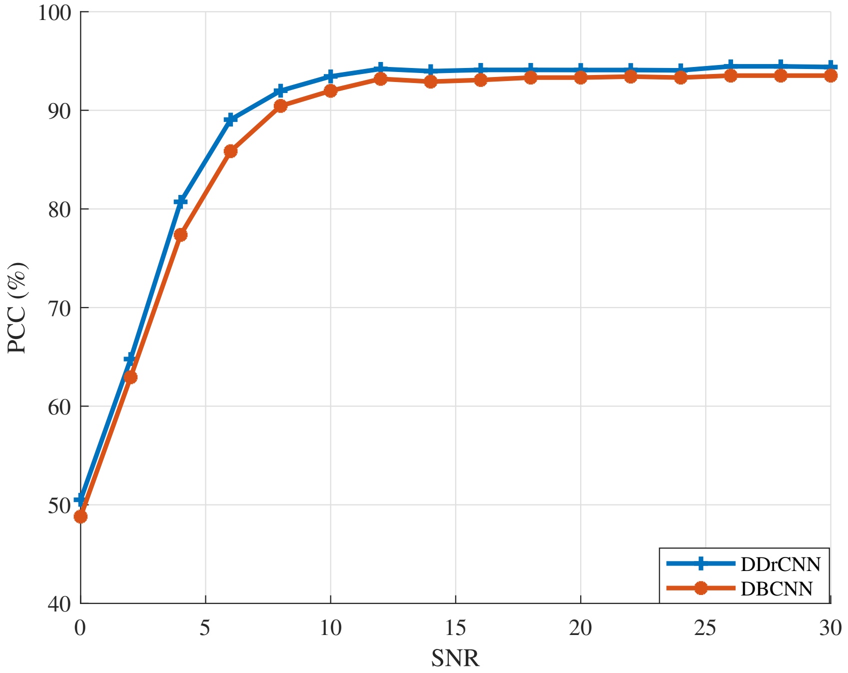



Figure 2.12: PCC performance of DDrCNN and DBCNN on RadioML dataset.

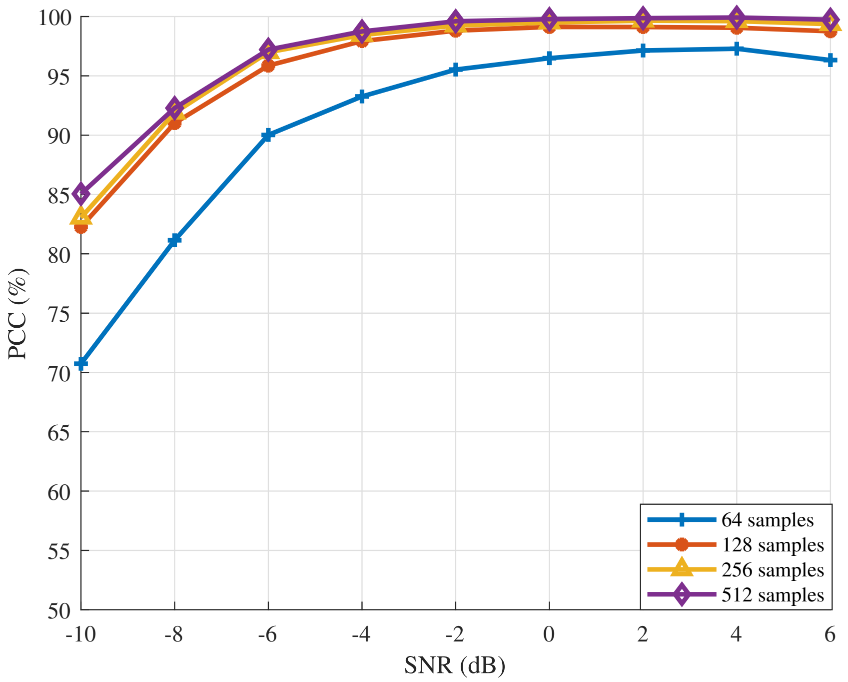



Figure 2.13: PCC performance of DDrCNN for different lengths of signals.

TH-4097_156102033

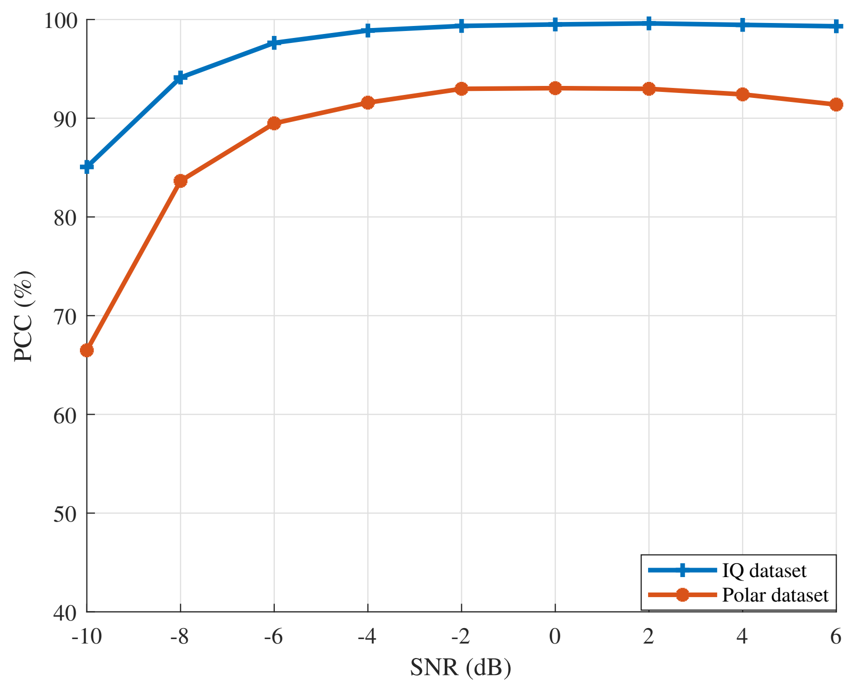
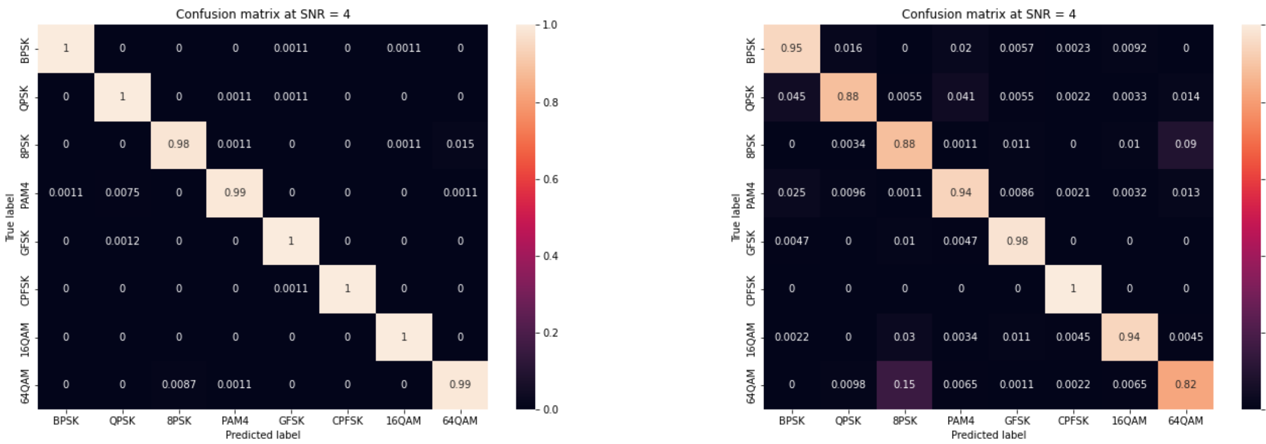



Figure 2.14: PCC performance of DDrCNN for Polar and IQ dataset.




(a) IQ data-set 4dB

(b) Polar data-set 4dB

Figure 2.15: Confusion matrix of DDrCNN for IQ data and Polar data.

TH-4097_156102033

### 2.4.3 The effect of the size and representation of the signals in the dataset on the PCC

The study by Wang et al. [25], recommended employing a signal length of 128 samples for each entry in the dataset. The data was encoded in IQ format during the dataset generation process. In this section, the appropriateness of selecting signals of length 128 samples and representing the same in IQ format is elaborated.

**Effect of length of signals in the dataset on the performance of DDrCNN:** To investigate the effect of changes in the number of samples per signal on the PCC, three more datasets were created, each containing 64, 256, and 512 samples per signal. These datasets were employed to train the DDrCNN, and the outcomes are graphically presented in Fig. 2.13, alongside the result obtained with 128 samples per signal. Though selecting a length of 64 samples per signal deteriorates the PCC, there is no considerable improvement for higher lengths like 256 and 512.

**Classification performance of DDrCNN on Polar dataset :** The performance of DDrCNN on a data set that represents the complex communication signal in polar form, as opposed to the IQ format, was explored. The polar dataset was created using MATLAB, following a method similar to IQ data generation. Instead of storing the in-phase and quadrature components as two separate rows, the polar dataset comprises two rows representing the magnitude and phase of the complex signal. The DDrCNN network is trained on the polar dataset, and the result compared with the IQ dataset is plotted in Fig. 2.14. The graph shows that the classification performance of the network is vastly superior when an IQ dataset is utilised. The confusion matrix at 4 dB SNR is given in Fig. 2.15, which depicts the difference in classification accuracy for the constituent modulation schemes.

The training performance and the classification results of both DDrCNN and DBCNN are compared to those of DrCNN in the succeeding section.

### 2.4.4 Performance comparison with DrCNN

It was reported that DrCNN outperforms state-of-the-art networks like RNN, DNN, and InceptionNet by a considerable margin on a data set generated through MATLAB when QAMs are kept as a single class [25]. In this section, the comparison between the DrCNN-based

TH-4097_156102033

Table 2.6: Comparison of number of trainable parameters

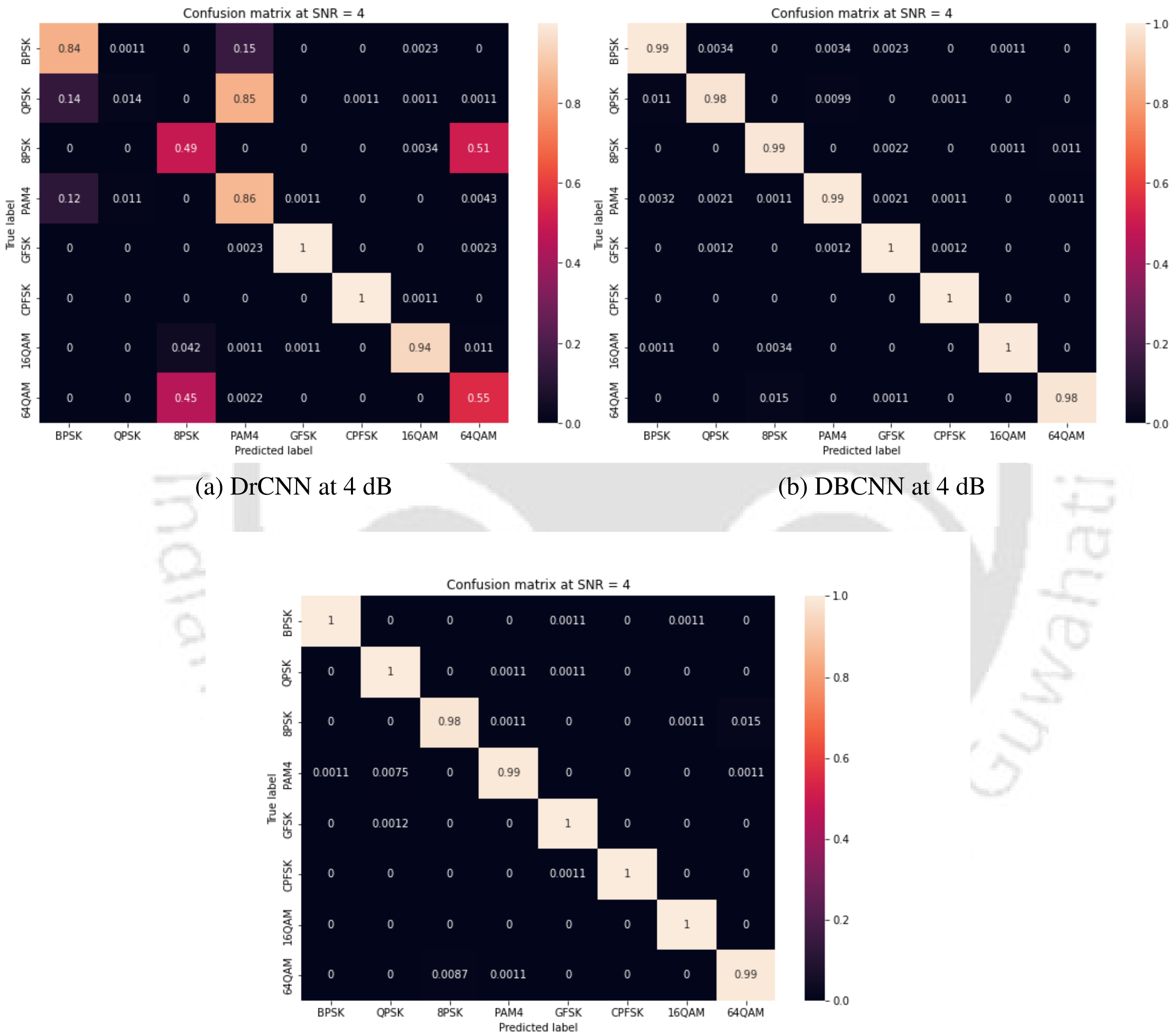
| Network | No. of Parameters |
|---|---|
| DDrCNN | 216,515 |
| DBCNN | 217,707 |
| DrCNN | 1,101,448 |

classifier and the proposed classifiers are presented. DrCNN was implemented as per the details given in the paper. Though the aim of DrCNN was not to classify the eight modulation schemes, it was for comparison because of its very similar structure. The networks were trained on the dataset, which had signals with a length of 128 samples.

DBNN and DDrCNN took 7 and 9 seconds per epoch, respectively, while training on the data with the same code parameters, and DrCNN took 11 seconds. For the dataset under consideration, DrCNN's classification performance was poor for 16-QAM vs. 64-QAM. Classification accuracy was plotted for these networks at various SNRs as shown in Fig.2.17. From the figure, one can see that both DDrCNN and DBCNN outperform DrCNN. For the dataset considered, DrCNN's classification performance was poor for QAMs as well as 8PSK/QPSK; hence, a significant difference in performance is seen in the confusion matrix plotted in Fig. 2.16.

The comparison of trained network parameters is presented in Table 2.6. It is evident from the Table that both DDrCNN and DBCNN have 21% of the trainable parameters compared to DrCNN. Notably, DrCNN requires an additional CNN for QAM classification. Hence, both DDrCNN and DBCNN emerge as superior classifiers in terms of complexity and performance.

The diagonal entries of the three confusion matrices in Fig. 2.16 are extracted and tabulated as the class-wise PCC in Table 2.7, along with the overall (mean) PCC, at SNR = 4 dB. The DrCNN exhibits a sharply non-uniform class-wise behaviour: the QPSK collapses to only 1.4% as it is predominantly misclassified as PAM4, while the 8PSK and 64-QAM pair is split almost evenly between the two classes. These failure modes bring down its overall PCC to 71.2%. In contrast, both the DBCNN and the DDrCNN classify every modulation class with a PCC of $\geq$ 98%, achieving an overall PCC of 99.1% and 99.5%, respectively, which is an improvement of about 28 percentage points over the DrCNN.




(a) DrCNN at 4 dB

(b) DBCNN at 4 dB

(c) DDrCNN at 4 dB

Figure 2.16: Comparison of the Confusion matrix of DrCNN, DBCNN and DDrCNN.

TH-4097_156102033

Table 2.7: Class-wise and overall percentage classification accuracy (PCC) of DrCNN, DBCNN and DDrCNN at SNR = 4 dB.

| **Modulation class** | **DrCNN (%)** | **DBCNN (%)** | **DDrCNN (%)** |
|---|---|---|---|
| BPSK | 84.0 | 99.0 | 100.0 |
| QPSK | 1.4 | 98.0 | 100.0 |
| 8PSK | 49.0 | 99.0 | 98.0 |
| GFSK | 100.0 | 100.0 | 100.0 |
| CPFSK | 100.0 | 100.0 | 100.0 |
| PAM4 | 86.0 | 99.0 | 99.0 |
| 16-QAM | 94.0 | 100.0 | 100.0 |
| 64-QAM | 55.0 | 98.0 | 99.0 |
| **Overall (mean) PCC** | **71.2** | **99.1** | **99.5** |

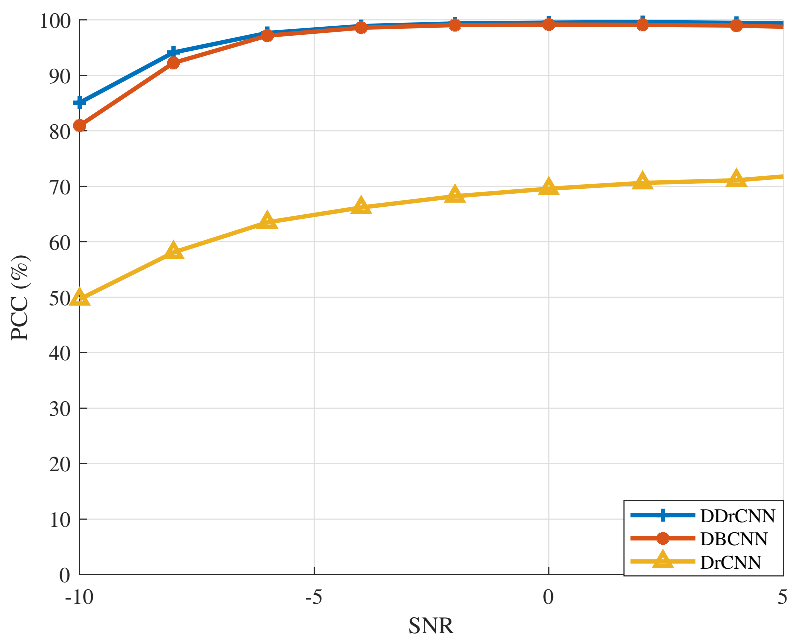



Figure 2.17: Performance comparison of DrCNN with proposed networks on the synthetic dataset.

## 2.5 Summary

This chapter, explored the AMC of 8 digital modulation classes by employing CNN-based DL classifiers. The state-of-the-art classifier for this problem employed a dual CNN configuration to categorise these eight modulation schemes effectively. Our findings demonstrate that eliminating dropout layers from the convolutional layers contributes to enhanced classification performance using a single SCNN only. By selectively applying dropouts or batch normalisation solely in the dense layers, the network size is reduced successfully. The resulting DDrCNN and DBCNN achieved an impressive accuracy of around 98% to 99% for SNRs above 0 dB for signals containing eight distinct digital modulation schemes for a pedestrian wireless channel having Rician fading. The validation of these results was conducted using the RadioML2018.01a dataset, confirming their reliability. DDrCNN is used as the classifier model in the subsequent chapters for its marginally superior performance.

TH-4097_156102033

# 3

# Cooperative Modulation Classification Employing SCNNs over Distributed Communication Networks

TH-4097_156102033

## 3.1 Introduction

Distributed communication systems are widely deployed in applications which demand resilience, scalability, and real-time responsiveness across geographically dispersed nodes. For example, in military applications, unmanned aerial vehicles are cooperatively utilised to perform signal detection tasks. In civilian domains, distributed communication frameworks are integral to wireless sensor networks and cognitive radio systems, where devices dynamically share spectrum information and adapt transmission strategies in real-time [39]. This helps in effectively utilising the spectrum and increasing the robustness and reliability. The Internet of Things (IoT) is an emerging distributed communication technology that is rapidly finding applications across diverse domains, including smart home appliances, industrial sensors, and healthcare systems [40, 41]. AMC plays a key role in IoT networks in ensuring security and efficiency.

A number of studies in AMC involve the utilisation of multiple receivers at the time of decision-making [42] [43]. Cooperating with multiple receivers in decision-making generally improves the performance of the classification system [44]. The cooperation of multiple nodes can be centralised or distributed. In the centralised architecture, the decisions or data from each sensor are transmitted to a fusion centre (FC) where the final decision is made, as shown in Fig. 3.1. In centralised architecture the FC will be capable to receive data/features and to process them to identify the modulation scheme. In the distributed model, each node processes its locally available data to generate an initial decision [45]. This decision is then refined through information exchanged with cooperating nodes. In an alternative scenario, an FC with limited computing capabilities may be employed solely to compile the inputs from the nodes and produce the final decision. The majority of the existing works fall in the first category of sensor cooperation in the case of AMC, and they utilised FB methods for classification purposes.

This chapter proposes to study the effect of cooperation in the case of SCNN-based AMC over multiple cooperating networks. In existing methods, the cooperating mechanisms use either higher-order cumulants (HOC) or likelihood-based methods for the classification [28, 46, 47]. In the case of HOC-based classification models, the class under consideration is dependent

TH-4097_156102033

on the selection of features. Hence, the number of modulation classes that can be identified is limited by the features selected for the classifier. The number of samples needed for the HOC-based classifier for effective classification is more than that of the DL-based classifiers. However, the DL-based classifiers are computationally more demanding than the conventional FB-based classifiers.

The previous chapter, proposed two SCNN architectures for the AMC of eight digital modulation schemes, demonstrating strong performance even at low SNRs. This work investigates how the DDrCNN classifier from our earlier study can be adapted for a cooperative communication scenario. Given that cooperation typically enhances AMC performance, the sample size of the CNN input is modified to further reduce the model's complexity. Such network trimming can facilitate the integration of these classifiers into hardware-constrained environments, such as IoT systems.

This chapter is organised as follows. A brief review of relevant literature is presented in Section 3.2. The proposed classification model is discussed in Section 3.3. Results for cooperative AMC utilising DL are elaborated in Section 3.4, followed by a summary in Section 3.5.

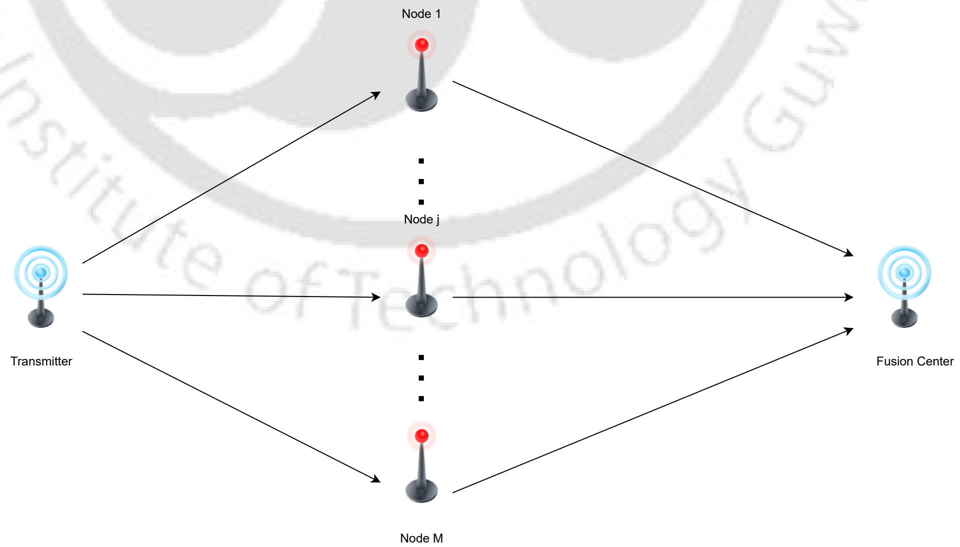



Figure 3.1: A distributed communication network with centralised fusion for decision cooperation.

TH-4097_156102033

## 3.2 Related Works on Cooperative AMC over Distributed Networks

The concept of cooperation is considered a successful solution to wireless communication problems where there exist multiple receivers or sensor nodes [48, 49]. In [42], the authors proposed an HOC-based cooperative automatic modulation classification (CAMC) under dispersive fading channels. Different fusion methods are discussed, and the effects of the number of nodes and channel estimation on the performance of AMC were studied. These methods include Hard Decision Fusion (HDF) [50] and Soft Decision Fusion (SDF) [51]. In HDF, each individual cooperative sensor or receiver node in a network performs a local AMC task and makes a local hard decision about the modulation scheme of an unknown signal. The SDF generalises the concept of optimal HDF. Instead of relying solely on the discrete local hard decisions made by individual nodes, SDF methods incorporate a ”soft decision vector (SDV)”. This SDV is employed as a measure of the reliability or conditional probability of the local decision. A critical aspect highlighted is the dependence of these fusion methods on predefined reference values that ideally match the actual channel conditions. In practical scenarios, obtaining perfect knowledge of these references is difficult, leading to mismatched references which can cause a considerable decline in AMC performance.

A detailed survey of modulation classification in communication systems employing multiple nodes is presented in [28] along with a new classifier based on estimated HOCs and LB-based decision cooperation. HOC features estimated by each sensor were transmitted to the fusion centre, where an LB combining algorithm is employed to obtain the final decision. Modulation schemes QPSK and 16QAM were considered for AMC. They carried out a detailed study on the effect of cooperation on AMC for both AWGN and Rayleigh fading channels, considering different numbers of receiver nodes. Since the classifier is based on cumulant features, the addition of modulation classes requires changes in the feature selection. This work assumes that the joint probability distribution of the cumulant features across the various nodes follows a multivariate Gaussian model. Hence, the likelihood-based decision combining depends on probabilistic models of the cumulant estimates, which require knowledge of their means and variances for different modulation schemes and SNRs. Differences between the estimates and

TH-4097_156102033

the actual values may affect the classification performance. In [52], the authors introduced CAMC using unmanned aerial vehicles. They employed a graph-based AMC method for local decisions and proposed novel weighted voting for decision fusion at the fusion centre. Again, this method might be affected by imperfect channel estimation, imperfect SNR estimation, and imperfect synchronisation in a practical scenario.

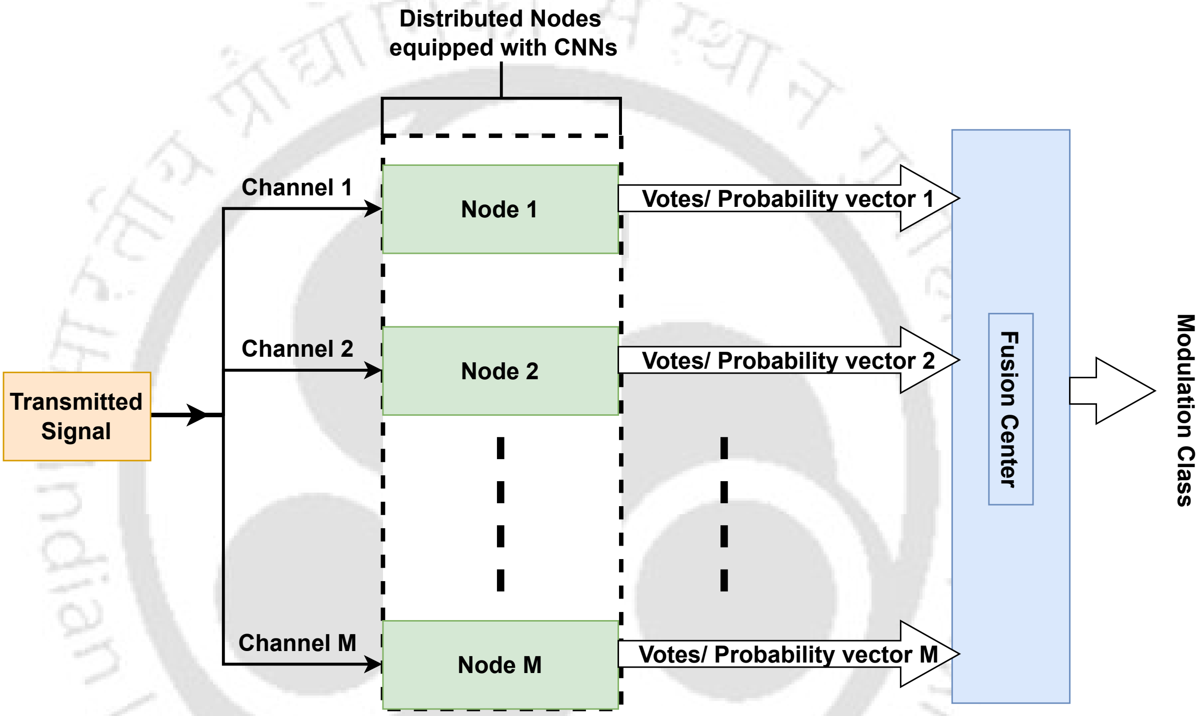



Figure 3.2: CNN-based system for CAMC over a distributed network.

## 3.3 System Model and Proposed Classifier

In contrast to the Rician channel considered in the preceding chapter, the current investigation assesses the AMC within a SISO channel subject to Rayleigh fading. This shift addresses a wireless environment characterised by the absence of a direct line-of-sight path, often posing greater challenges for communication systems. In this work, it is assumed that CNN classifiers are present at each node of the distributed network. These nodes are capable of communicating flawlessly with the fusion centre, where the final decision is made. At the sensor node, the trained DL networks will process the input signal, transmit the probability linked with each output class to the fusion centre for cooperative decision-making. The overall classification process is depicted in Fig. 3.2.

TH-4097_156102033

### 3.3.1 CNN used for classification

For this work, DDrCNN has been adopted for identifying the modulation classes at each node in the communication network as given in Figure 3.3. The detailed architecture and performance evaluation of this network were discussed in Chapter 2. The same network is selected to depict the improvement in PCC when cooperative rules are applied. The DDrCNN has two convolution layers that function as the feature extraction unit. The first layer contains 128 filters of kernel size (2,6) with the rectified linear unit (ReLU) as the activation function. The second convolutional layers consist of 64 filters with a kernel size of (1,14). Two hidden layers with 128 and 64 filters are employed with ReLU as the activation function in the dense layers. Dropout are added after each hidden layer. The output layer is a softmax layer.

Categorical cross-entropy, which is defined based on the ground truth of class symbols represented in one-hot encoding ( $y_i$) and their corresponding predictions ($\hat{y}_i$), is chosen as the loss function and is given by.

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(\hat{y}_i) \tag{3.1}$$

where N is the number of samples considered in one training batch.

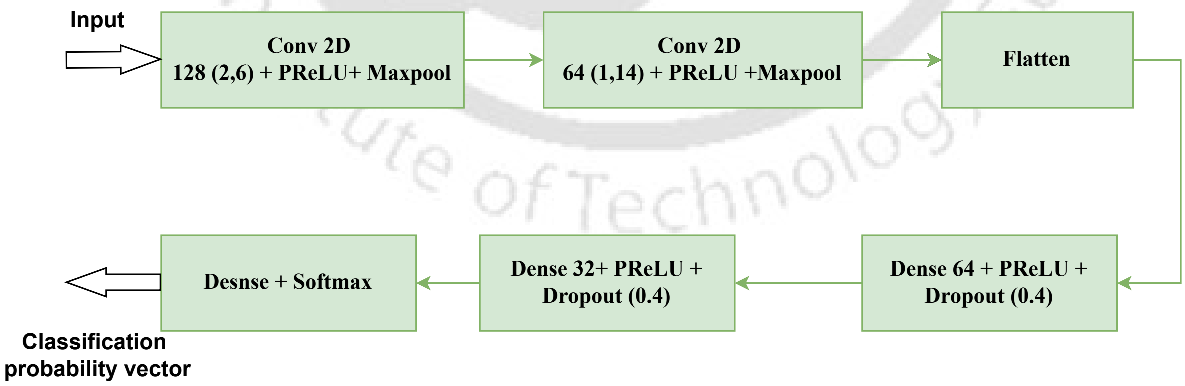



Figure 3.3: Architecture of CNN employed at each node of the distributed network.

### 3.3.2 CAMC over Distributed SISO Networks

In this work, the nodes in the network are assumed to be SISO. The nodes equipped with trained CNNs will transmit the classification probabilities of the constituent modulation schemes.

TH-4097_156102033

At the FC, a cooperative decision is implemented to combine the decisions from each node. Voting-based and averaging-based decision cooperation methods are considered. While the averaging method computes the average of the probabilities obtained by all of the receiving antennas and follows the rule that the modulation type with the highest probability is the final predicted modulation type, the voting method is based on the count of individual modulation types provided by each antenna, and the modulation type with the majority vote is the selected one. The algorithms for both methods are mentioned in Algorithm 3.1 and Algorithm 3.2.

The notations used in the cooperative system model and in Algorithms 3.1 and 3.2 are summarised in Table 3.1.

Table 3.1: Notations used in the cooperative system model and in Algorithms 3.1 and 3.2.

| **Symbol** | **Definition** |
|---|---|
| $\mathcal{M}$ | Pool (set) of candidate modulation classes considered for classification. |
| $\lvert\mathcal{M}\rvert$ | Cardinality of $\mathcal{M}$, i.e. the total number of modulation classes. |
| $m_n$ | The $n^{\text{th}}$ modulation class in $\mathcal{M}$, $n \in \{1, 2, \ldots, \lvert\mathcal{M}\rvert\}$. |
| $M$ | Total number of cooperating sensor nodes in the distributed network. |
| $\mathcal{N}^j$ | The $j^{\text{th}}$ sensor node, $j = 1, 2, \ldots, M$, equipped with a trained DDr-CNN classifier. |
| $Y^j$ | Received in-phase/quadrature signal observed at node $\mathcal{N}^j$ over the observation window of 128 samples (input to the CNN at that node). |
| $P(m_n \lvert Y^j)$ | Softmax output of the CNN at node $\mathcal{N}^j$ for class $m_n$ given the received signal $Y^j$; interpreted as the posterior probability that the transmitted modulation is $m_n$. |
| $\boldsymbol{P}^j$ | Soft-decision probability vector output by the CNN at node $\mathcal{N}^j$: $\boldsymbol{P}^j = [P(m_1 \lvert Y^j), P(m_2 \lvert Y^j), \ldots, P(m_{\lvert\mathcal{M}\rvert} \lvert Y^j)]$. |
| $\hat{m}_n^j$ | Hard decision (predicted modulation class) at node $\mathcal{N}^j$: $\hat{m}_n^j = \arg\max_n P(m_n \lvert Y^j)$. The subscript $n$ records the index of the modulation class selected at node $\mathcal{N}^j$, i.e. $\hat{m}_n^j = m_n$ for that specific $n$; it is not a free running index. |
| $\overline{\boldsymbol{P}}$ | Element-wise mean of the probability vectors $\{\boldsymbol{P}^j\}_{j=1}^{M}$ computed at the fusion centre: $\overline{\boldsymbol{P}} = \frac{1}{M}\sum_{j=1}^{M} \boldsymbol{P}^j$. |
| $\overline{P}(n)$ | The $n^{\text{th}}$ element of $\overline{\boldsymbol{P}}$. |
| VoteCount$(n)$ | Number of nodes among $\{\mathcal{N}^j\}_{j=1}^{M}$ whose hard decision $\hat{m}_n^j$ corresponds to the $n^{\text{th}}$ modulation class $m_n$. |
| $\hat{m}$ | Final cooperative decision at the fusion centre after applying the voting (Algorithm 3.1) or averaging (Algorithm 3.2) rule. |

TH-4097_156102033

**Algorithm 3.1:** Cooperative AMC for cooperating nodes – voting method.

**Input:** Votes from each of the nodes $\mathcal{N}^j$, $j = 1 \rightarrow M$

**Output:** Selected modulation type $\hat{m} \in \mathcal{M}$

1 For $j = 1 \rightarrow M$, input votes for each modulation scheme, $\hat{m}_n{}^j = \arg\max_n P(m_n|Y^j)$ from each CNN node.

2 Count the number of occurrences of each modulation class in the predictions to get the vote count(n), $n \in [1, |\mathcal{M}|]$.

3 Identify the modulation class with the highest vote count: $\hat{m} = \underset{n\in[1,|\mathcal{M}|]}{\text{argmax}}\,(\text{VoteCount}(n))$

**Algorithm 3.2:** Cooperative AMC for cooperating nodes - averaging method.

**Input:** CNN decision probabilities from each of the nodes $\mathcal{N}^j$, $j = 1 \rightarrow M$

**Output:** Selected modulation type $\hat{m} \in \mathcal{M}$

1 Input the softmax probability vectors, $\boldsymbol{P}^j = [P(m_1|Y^j), P(m_2|Y^j), ..., P(m_{|\mathcal{M}|}|Y^j)]$ from the CNNs for $j = 1 \rightarrow M$.

2 Compute the average probability vector at the receiver: $\overline{\boldsymbol{P}} = \frac{1}{M}\sum_{j=1}^{M} P^j$

3 Select the modulation type with the highest averaged probability: $\hat{m} = \underset{n\in[1,|\mathcal{M}|]}{\text{argmax}}(\overline{P}(n))$

To implement the cooperative scheme, the trained DDrCNN is used for detecting the modulation class at each node. The output of the classifier is the softmax probability vector. Now, to test the effectiveness of the CAMC, test data is generated for each of the nodes and passed on to the classifier. The outputs corresponding to each of the test inputs are combined using the cooperative decision rule for the final decision.

## 3.4 Simulation Results

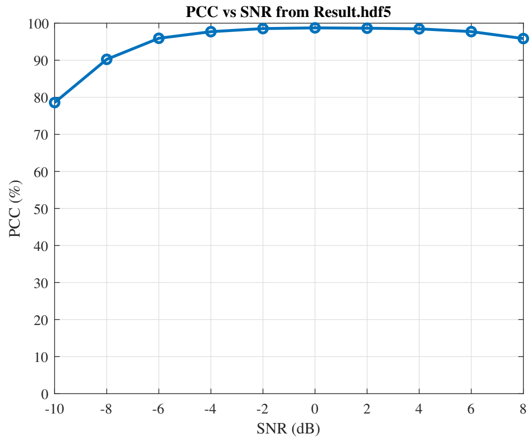
The performance of CAMC is tested using the datasets generated using MATLAB, following the signal model explained earlier. The modulation classes considered are BPSK, QPSK, 8PSK, GFSK, CPFSK, PAM4, 16QAM, and 64QAM. The dataset for training the CNN classifier employed in each of the distributed nodes is generated using Matlab. The SNR ranges from -10 dB to 10 dB for the training dataset. The adaptive moment estimation (Adam) optimiser is used to minimise the loss function. The learning rate value employed for CNN training is 0.001. The CNN was trained for 400 epochs, and an early stopping callback of patience 20 was used. The batch size is set to 256 during training. The DDrCNN was trained and tested under two different scenarios with varying input signal sizes. In the first scenario, consistent with previous work, the dataset contained signals of size 2×128, while in the second scenario, the dataset included signals of size 2×64.

For the training and testing of CNNs, a computing machine with NVIDIA GeForce GTX 1650Ti, offering 4 GB RAM, is utilised. Deep learning modules are implemented using Keras with TensorFlow as the back end.

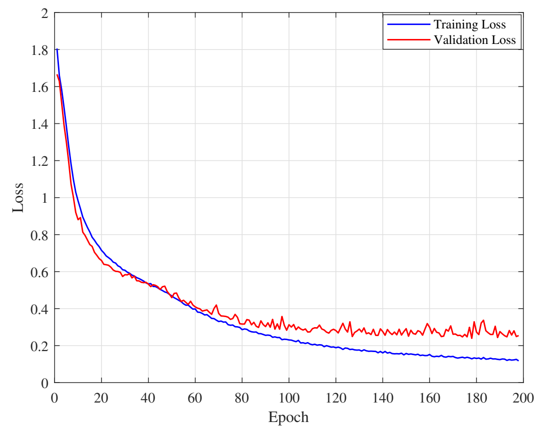
### 3.4.1 Simulation results with signal size 2x128

The training converged in 198 epochs with an overall 94% correct predictions. Maximum PCC of around 97% is achieved at an SNR of 2 dB. The training performance of the classifier is plotted in Fig. 3.4. The validation loss for the network appeared to become constant after 120 epochs, suggesting over-fitting.

TH-4097_156102033




(a) Classification performance




(b) Training loss vs Validation loss

Figure 3.4: Model training behaviour and classification performance for a single node (2×128 input format)

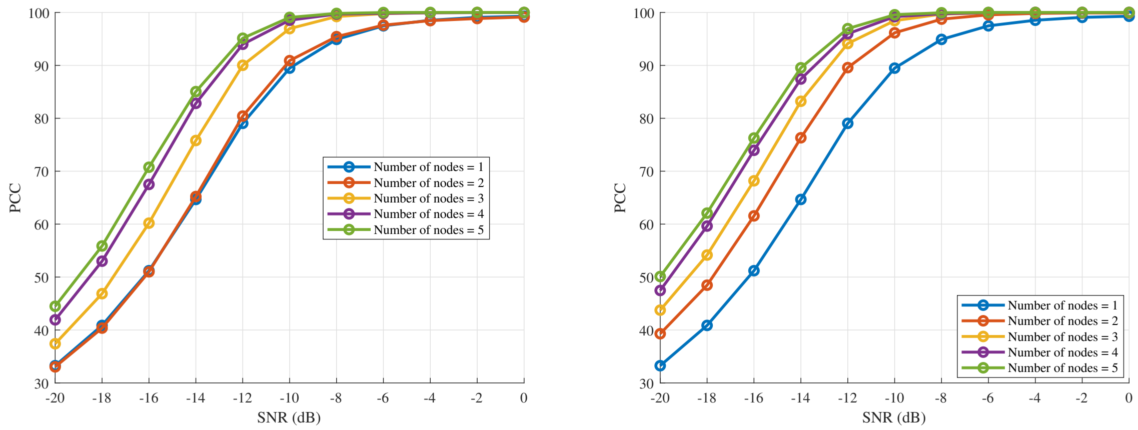
To study the effect of cooperation employing multiple nodes available in the network, the number of nodes considered were 1,2,3,4 and 5. A separate test dataset, which is not a subset of the training dataset, is generated and assigned to each node. For the testing dataset, the SNR varied between -20 dB and 0 dB since the classification accuracy is high from $-10$ dB itself. Both the cooperation rules mentioned in Section 3.3.2 were employed using this dataset. It was observed that both methods improved the classification performance, but the averaging-based cooperation performed better than the voting-based cooperation at lower SNRs. The PCC for both methods considering multiple numbers of nodes is given in Fig. 3.5. A comparison plot between averaging and voting-based cooperative AMC is included as Fig. 3.6.

TH-4097_156102033

### 3.4.2 Simulation results with signal size 2x64.

The objective of reducing the sample observation length is to accelerate the processing and to design a more compact classifier. Table 3.2 shows the details of DDrCNN while a compact input is used. While training the 64 samples input, DDrCNN converged in 332 epochs. The maximum PCC obtained is around 88% at 8 dB. The training performance is plotted in Fig. 3.7.

A test dataset is generated for input signals of 64 samples, with SNR values ranging from $-10$ dB to 10 dB. In this case, the number of nodes considered are 1,2,3,4,5,10. Table 3.3 presents the classification accuracies obtained for an input size of 64 samples with cooperation enabled. It can be observed that the accuracy reaches nearly 98% even at an SNR of $-4$ dB when employing 10 cooperating nodes. The network size of the DDrCNN for an input size of 128 samples is approximately 230K parameters, whereas for an input size of 64 samples, it is about 160K parameters.




(a) Voting based cooperation (b) Averaging based cooperation.

Figure 3.5: PCC for CAMC with multiple SISO nodes for input size 2x128

TH-4097_156102033

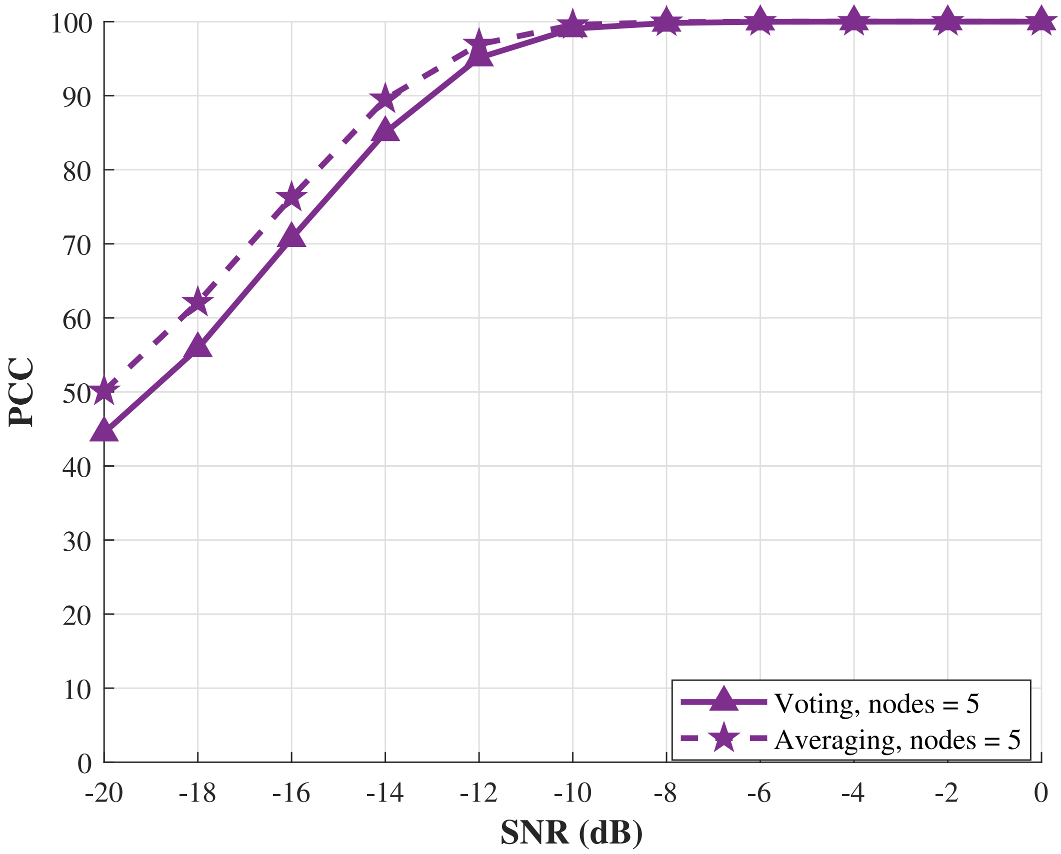
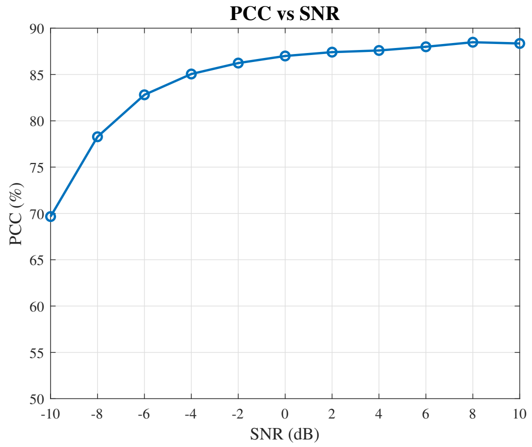



Figure 3.6: Accuracy comparison of CAMC: Averaging vs. Voting (Input Size: 2×128)

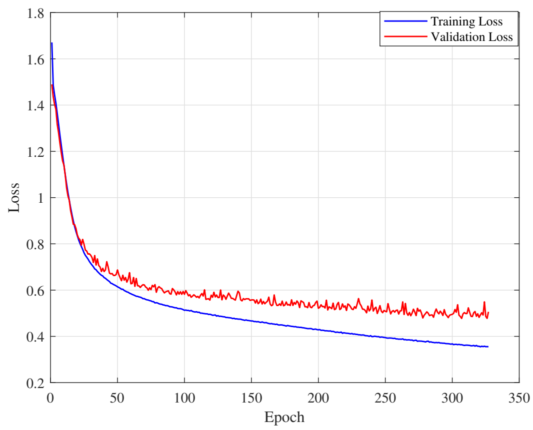
Table 3.2: Layers of DDrCNN for input signal size 2x64

| **Layer** | **Output dimensions** |
|---|---|
| Input | (2,128) |
| Convolution2D (Filters 128, (2,6)) | ( 59, 128) |
| MaxPooling2D | (30, 128) |
| PReLU | (30, 128) |
| Convolution2D (Filters 64, (1,14)) | (17, 64) |
| Maxpooling2D | (9, 64) |
| PReLU | (9, 64) |
| Flatten | 576 |
| Dense + PReLU | 64 |
| Dense + PReLU | 32 |
| Dense + Softmax | Number of classes |

TH-4097_156102033




(a) Classification performance




(b) Training loss vs Validation loss

Figure 3.7: Model training behaviour and classification performance for a single node (2×64 input format)

Table 3.3: PCC comparison between single node and averaging-based fusion with 10 nodes for 64 samples input

| SNR (dB) | Single Node (%) | Averaging-based CAMC (10 Nodes) (%) |
|---|---|---|
| −10 | 69.66 | 84.31 |
| −8 | 78.28 | 92.28 |
| −6 | 82.81 | 96.79 |
| −4 | 85.05 | 98.33 |
| −2 | 86.24 | 98.33 |
| 0 | 87.00 | 98.45 |
| 2 | 87.41 | 98.47 |
| 4 | 87.60 | 98.56 |
| 6 | 87.99 | 98.67 |
| 8 | 88.49 | 98.70 |
| 10 | 88.35 | 98.68 |

TH-4097_156102033

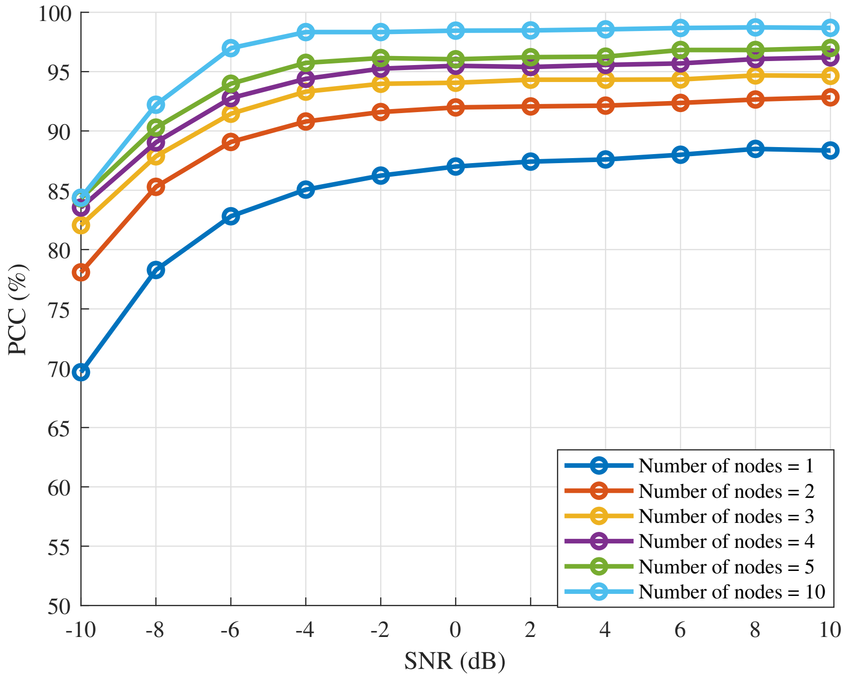



Figure 3.8: PCC for CAMC with multiple SISO nodes for input size 2x64

## 3.5 Summary

In this study, the efficacy of CAMC in Single-Input Single-Output (SISO) channels was demonstrated using a deep learning framework. The system architecture employed an SCNN-based decision generator at each cooperating node, with a decision combiner at the fusion centre to aggregate outputs. The cooperative modulation classifier was implemented and evaluated on a locally generated dataset in MATLAB for both 64 and 128 sample input lengths. Results showed that cooperation among multiple nodes significantly enhanced AMC performance, achieving nearly 98% accuracy at an SNR of $-4$dB with 10 cooperating nodes for the 64-sample case. The DDrCNN model used in this work had a network size of approximately 230K parameters for 128 sample inputs and about 160K parameters for 64 sample inputs, indicating that high classification performance can be achieved even with reduced model complexity.

TH-4097_156102033

# 4

# AMC of MIMO Signals using Shallow CNNs

TH-4097_156102033

## 4.1 Introduction

The previous chapters examined the AMC of signals over the SISO channel using an SCNN classifier, which achieved higher classification accuracy than the FB classifier. In distributed SISO systems, cooperation among multiple nodes helped to improve the classification accuracy. Many of the recently deployed communication systems, such as cellular networks, WiFi, and satellite communication systems, employ multiple-input multiple-output (MIMO) technology for improved channel utilisation and increased reliability. Consequently, research on AMC for MIMO systems has gained significant attention. There is a scope to extend the proposed SCNN classifiers for MIMO signals.

A number of works in the literature have employed both the LB and the FB methods for the AMC of MIMO signals. In [53], Choqueuse et al. have introduced an LB algorithm for the AMC of MIMO signals under uncorrelated Rayleigh channel assumptions. In their work, two LB classifiers were considered for solving the problem, namely, ALRT and HLRT. For the HLRT classifier, they applied independent component analysis and the phase correction algorithm for channel estimation. A feature fusion-based method was introduced in [54] for the classification considering a known MIMO channel matrix. In this work, the authors have proposed an FB technique based on higher-order moments and cumulants and employed an artificial neural network (ANN) as the classifier. The classification results are presented for both known channel state information (CSI) and unknown CSI for a correlated MIMO channel. Another FB classification method for MIMO systems under frequency-selective fading channels is presented in [55], which utilises the peaks in the cross-correlation function for differentiating the modulation formats. Zhu and Nandi [56] have employed the expectation-maximisation algorithm to estimate the MIMO channel matrix, and then an LB-based method is employed for the AMC.

The first work on classifying MIMO signals over the Rayleigh fading channel using a CNN-based classifier was proposed in [29]. The authors have employed a zero-forcing equaliser for CSI estimation and used the estimated signals to train the CNN. The performance of the classifier under imperfect CSI was also studied in the same work. The requirement for CSI at the

TH-4097_156102033

receiver renders this approach unsuitable for blind modulation classification systems. Another CNN-based cooperative AMC method for the MIMO signals over the Rayleigh fading channel was proposed in [30]. In both of these works, the CNN included one-dimensional convolutional layers. In the latter work, the authors had done a detailed analysis of the effect of cooperating with the CNN's decision at each receiver antenna. The AMC results for cooperation were presented based on the voting and averaging methods. The results show that averaging-based methods outperformed voting-based methods. This work motivated us to evaluate the performance of the networks employed in the previous chapter on the MIMO dataset. Additionally, the possibility of applying feature-fusion-based cooperation on the received MIMO signals is explored.

In the majority of research existing in the literature, the MIMO channel models assume a rich scattering environment. In practical terms, MIMO systems may function in environments with insufficient scattering, potentially resulting in channels with reduced ranks [57, 58]. The keyhole channel, which is the extreme case of rank-deficient channels, when modelled mathematically, appears to be double Rayleigh distributed [59]. The channel matrix will be rank deficient, and the AMC for MIMO signals under such channels is addressed in [31] using an FB approach. This method is able to discriminate only lower-order PSK constellations under unknown CSI. To the best of our knowledge, the investigation of DL-based AMC under a rank-deficient channel has not been carried out in the literature.

The problems addressed in this chapter can be summarised as follows:

(i) The proposed DDrCNN and DBCNN-based classifiers for the SISO signals are extended for the AMC over MIMO Rayleigh channels. A modified DDrCNN classifier is proposed for the AMC of MIMO signals.

(ii) The decision fusion and feature fusion-based cooperative methods at the MIMO receiver are evaluated.

(iii) The AMC performance of a few competing CNN classifiers is compared with the proposed one. The effectiveness of the CNN-based classifier in the presence of antenna

TH-4097_156102033

correlation is also studied.

(iv) The effectiveness of the CNN-based classifier for the AMC of MIMO signals over keyhole channels is investigated. The work includes (i) assessing its performance by training the CNN with signals over MIMO keyhole channels, (ii) analysing the generalisation capability of the network when trained on different datasets, and (iii) comparing its classification accuracy with that of existing FB classifiers under identical channel conditions.

The rest of the chapter is organised as follows. In Section 4.2, the channel model for MIMO signals is discussed. Section 4.3 explains the SCNN architecture proposed in this work and the cooperative methods employed at the MIMO receiver for improving AMC performance. Competing CNN-classifiers selected for comparison are covered in Section 4.3.2. Section 4.4 details the simulation results of the AMC over MIMO Rayleigh channel. Effectiveness of modified DDrCNN on AMC of signal over keyhole channel discussed in Section 4.5. A summary of the chapter is presented in Section 4.6.

## 4.2 AMC of MIMO Signals Over Rayleigh Channel

This section presents the CNN-based AMC of MIMO signals over a Rayleigh fading channel. The performance of DBCNN and DDrCNN architectures is evaluated to assess their effectiveness in the MIMO scenario. Furthermore, a modified version of the DDrCNN is proposed. The channel model adopted for this study is described below.

### 4.2.1 MIMO Rayleigh Channel Model

In this section, the AMC of MIMO signals over a Rayleigh fading channel using SCNNs is discussed. For the SISO scenario, presented in the earlier chapter, the SCNN classifier was trained with the received signal, over a Rician channel. For this study, a time-invariant, block-fading MIMO channel with $N_T$ transmitting and $N_R$ receiving antennas is selected. Hence at any instance $k$, the input signal vector $\boldsymbol{X}(k)$ and the output signal $\boldsymbol{Y}(k)$ are related by,

$$\boldsymbol{Y}(k) = \boldsymbol{H}\boldsymbol{X}(k) + \boldsymbol{W}(k) \tag{4.1}$$

where $\boldsymbol{H}$ is the complex MIMO channel matrix of size $N_R \times N_T$, and $\mathbf{W}(k)$ is the $N_R \times 1$ zero

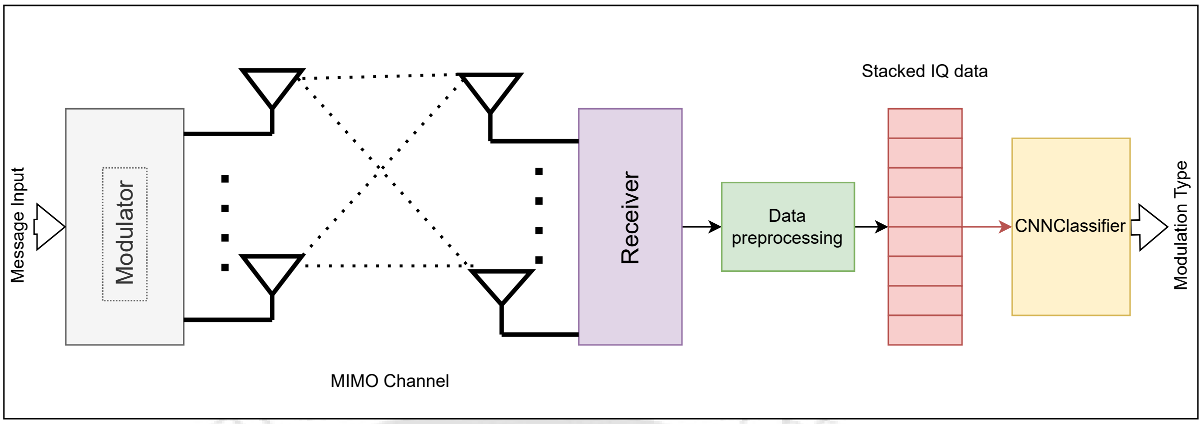



Figure 4.1: The structure of CNN-based AMC of MIMO signals - non-cooperative method.

mean circularly symmetric complex Gaussian noise vector.

In many cases, correlations exist between the signals over a MIMO channel primarily due to the spatial arrangement of multiple antennas. The correlations are captured in terms of the transmit correlation matrix $\boldsymbol{\theta_T}$ and the receive correlation matrix $\boldsymbol{\theta_R}$ and combined using the Kronecker model [60] to get the channel matrix as $\mathbf{H}$ [57, 61],

$$\boldsymbol{H} = \boldsymbol{\theta_R^{1/2} H_R \theta_T^{1/2}} \tag{4.2}$$

where $\boldsymbol{H_R}$ is the uncorrelated MIMO channel. The spatially correlated channel considered here follows the Kronecker model [60]. The elements of the correlation matrix $\boldsymbol{\theta}$ are given by,

$$[\boldsymbol{\theta}]_{i,j} = \begin{cases} \rho^{j-i}, & i \leq j \\ [\boldsymbol{\theta}]^*_{j,i}, & i > j \end{cases} \tag{4.3}$$

where $\rho$ is the complex correlation coefficient of the neighbouring antennas. The two correlation matrices, $\boldsymbol{\theta_R}$, and $\boldsymbol{\theta_T}$, are generated using the receiver and transmitter antenna correlations $\rho_r$ and $\rho_t$ respectively according to (4.3).

In a MIMO scenario, the system is equipped with multiple receiving antennas. Hence, the initial approach can be modelling the input as a stack of IQ signals received from each of the antennas. The size of the stacked IQ vectors at the MIMO receiver is $2N_R \times 128$. The CNN classifier treats this as an image of the same size. This method for MIMO signals classification

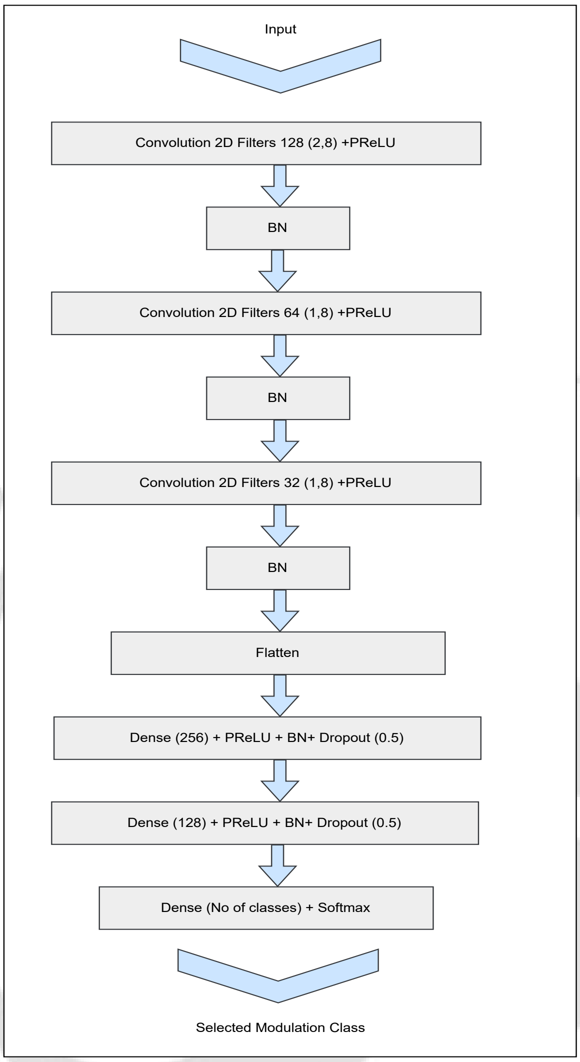



Figure 4.2: Structure of the modified DDrCNN.

will be mentioned as non-cooperative AMC. A pictorial representation is given in Fig. 4.1. In the case of AMC employing cooperative methods, the network's input size is different as explained in section 4.4.1.

A modified version of DDrCNN is utilised as the CNN for the classification of MIMO signals. Details of the employed CNN are given in the following Section.

## 4.3 Proposed SCNN Architecture for the AMC of MIMO Signals

The CNNs employed in the AMC problems in the previous two chapters include two convolutional layers and three dense layers. Since AMC classifiers are to be deployed in environments with limited computational capabilities, the size of the DL classifier should be small. In this work, a modified version of DDrCNN in the previous chapters is utilised as the classifier.

TH-4097_156102033

By removing pooling layers and dropouts from the convolutional layers, the CNN is shown to achieve better AMC performance in the case of SISO signals [62]. In most of the CNN-AMC methods, the network is able to achieve significantly better results than the FB-based methods with fewer symbols at the receiver. Since the number of symbols in the received signal is smaller, the CNN size remains manageable even with the pooling layers removed. The removal of pooling layers and dropouts from the convolutional layers allows the CNN to keep the richness of the features and helps to achieve better classification results.

Since the CSI is unknown at the receiver, employing the same CNN architecture may not be suitable for the DL-AMC of MIMO signals. The hidden layer size shall be increased to handle the effects of multipath signals received at each of the MIMO antennas. The architecture of DDrCNN proposed for AMC under the MIMO Rayleigh channel is presented in Fig. 4.2. Three convolution layers function as the feature extractor. The first layer contains 128 filters of kernel size (2,8) with PReLU as the activation function. The second and third convolutional layers consist of 64 and 32 filters, respectively, with a kernel size of (1,8). Two hidden layers with 256 and 128 nodes are employed with ReLU as the activation function in the dense layers. BN and dropout are added after each of the hidden layers. The output layer is a softmax layer with the number of output nodes equal to the modulation classes. This network is further referred to as the modified DDrCNN.

The network is trained using the corresponding dataset for both non-cooperative and co-operative cases. The categorical cross-entropy between the ground truth of the class symbols represented in one-hot encoding ($\boldsymbol{y}_i$) and its prediction ($\hat{\boldsymbol{y}}_i$) is used as the loss function. It is given by,

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{y}_i \cdot log(\hat{\boldsymbol{y}}_i) \tag{4.4}$$

where N is the number of samples considered in one training batch.

The proposed network and the networks mentioned in the previous chapter are trained and tested on the MIMO Rayleigh data set. Cooperative methods are applied to improve classification accuracy further.

TH-4097_156102033

### 4.3.1 CAMC of MIMO Signals for Performance Improvement

The literature shows that when there are multiple antennas at the receiver, one can cooperatively combine the predictions at each of them to improve the overall AMC performance of the system [30, 43, 61]. In cooperative AMC, decision-fusion-based cooperation was compared with feature-fusion-based cooperation. The details of each method are provided below.

**Decision fusion based CAMC**

The schematic representation of cooperative AMC using decision fusion is shown in Fig. 4.3. The CNN is trained with the output from each of the receiving antennas with proper labels. At the time of evaluation, the decisions from the receiving antennas are combined to predict the correct modulation format. In [30], the authors have shown that direct averaging-based cooperation outperforms the direct voting method in terms of classification accuracy for various antenna configurations ($N_T \times N_R$) like $1 \times 4$, $2 \times 4$ and $4 \times 4$ . A similar result is established in our work on CAMC for a distributed network also. Hence, in this work, an averaging-based cooperation to combine the results from each antenna is applied. For combining the decisions, the same algorithm (3.2) applied for averaging-based cooperation employed in CAMC for a distributed communication network has been selected.

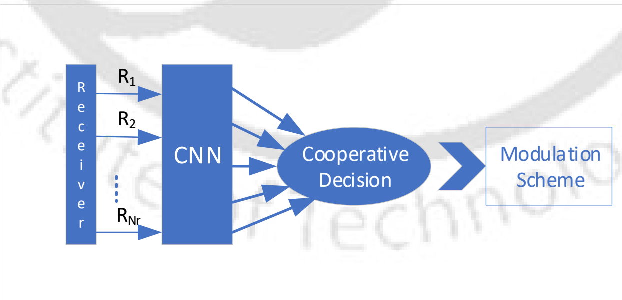



Figure 4.3: Schematic diagram of the AMC system employing decision fusion cooperation.

**Feature fusion-based CAMC**

As an alternative to the decision cooperation at the output of the CNN, a feature-fusion method is proposed here. The features obtained from the last hidden layers of the modified DDrCNN after training are fused together to form a new input signal for an ANN classifier. A schematic diagram of the process is shown in Fig. 4.4.

TH-4097_156102033

The ANN classifier selected for classification consists of two fully connected layers. The final dense layer of DDrCNN has a size of 128. Hence, the input data shape for the ANN classifier shall be $N_R$ times 128. The two hidden layers of the ANN classifier consist of 128 and 64 neurons, respectively. BN and dropouts are added after each hidden layer. A softmax output layer is employed to select the output class. The categorical cross-entropy in (4.4) is selected as the loss function and Adam as the optimiser with a learning rate of 0.01.

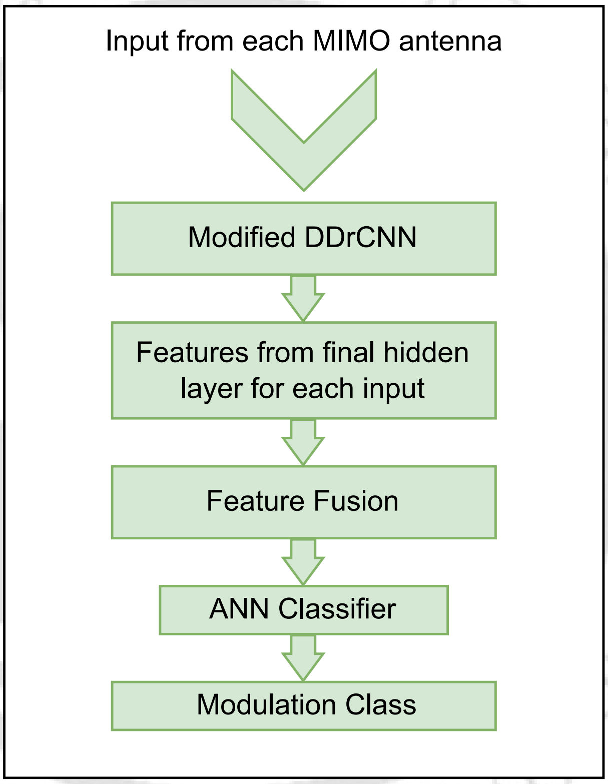



Figure 4.4: Schematic diagram of the AMC system employing feature fusion.

### 4.3.2 Other Competing CNNs Selected for Comparison

To compare the performance of the proposed classifier, three CNN architectures from existing works are selected. Out of these, the first two CNN architectures (Maxpool and Dropout CNN) are well discussed by the DL-based AMC community in the SISO scenario. The architecture of DrCNN is presented in Chapter 2. The third network architecture (1DCNN) is the pioneering work on the DL-based AMC for MIMO signals [29]. A comparison of the classification performances of the networks is presented in the results section. Table 4.1 outlines the architecture of MaxpoolCNN and 1DCNN.

TH-4097_156102033

Table 4.1: Architectures of Maxpool CNN and 1D CNN

| **Layer** | **Maxpool CNN** | **1D CNN** |
|---|---|---|
| Input | $2 \times 128$ | $2 \times 128$ |
| Conv 1 | 128 filters, kernel $2 \times 8$, ReLU | Conv1D(128, 8) + BN + ReLU + Dropout(0.5) |
| MaxPool 1 | Pool size 2, stride 2 | — |
| Conv 2 | 64 filters, kernel $1 \times 16$, ReLU, | Conv1D(64, 4) + BN + ReLU + Dropout(0.5) |
| MaxPool 2 | Pool size 2, stride 2 | — |
| Flatten | — | — |
| FC 1 | Dense(128) + ReLU | Dense(256) + BN + ReLU + Dropout(0.5) |
| FC 2 | Dense(64) + ReLU | Dense(128) + BN + ReLU + Dropout(0.5) |
| FC 3 | Dense(32) + ReLU | — |
| Output | Dense(number of classes) + Softmax | Dense(number of classes) + Softmax |

## 4.4 Results and discussions

The superiority of modified DDrCNN over DBCNN and DDrCNN for MIMO signals is established in the first result section. Later, detailed performance simulation results for the modified DDrCNN-based classifier, along with three other CNN-based DL classifiers described in the previous Section, were also presented. The results for the AMC performance evaluation under MIMO keyhole channel are presented in sub-section 4.5.3. The details of the data generation and implementation are presented first, followed by the classification results.

### 4.4.1 Generation of the MIMO Dataset

To the best of the current knowledge, there is currently no publicly available dataset for the performance evaluation of MIMO AMC. The dataset used for training and testing the proposed classifier is generated using the help of DL toolbox in MATLAB. For this work, a data set is created, according to the model described in (4.1), comprising five modulation types: BPSK, QPSK, OQPSK, 8-PSK and 16-QAM. Initially, a random sequence of length N is generated, which is then modulated to produce the complex modulated vector $\mathbf{X}$ of the same size. The vector $\mathbf{X}$ is normalised to unit power and reshaped into $N_t \times N/N_t$, where $N_t$ is the number of transmitting antennas of the MIMO system. The signal is then passed through the Rayleigh fading channel and received at each of the receiving antennas. The complex baseband signal received at a particular antenna $j$ can be expressed as $\boldsymbol{y}^j = [y_1(j)\, y_2(j) ... y_{N/N_t}(j)]^T$, where $j \in [1, N_r]$.

TH-4097_156102033

**(a) Dataset for Non-cooperative MIMO AMC**

To generate the dataset for training and testing the CNN for the non-cooperative case, the real and imaginary parts of the signals at each receive antenna are combined to form a sample (size $2N_r \times N/N_t$) of the dataset. The training dataset contains 20000 signals per SNR per modulation type, while the testing dataset contains 2000 signals per SNR per modulation type.

**(b) Dataset for Cooperative MIMO AMC**

In the case of cooperative AMC, the real and imaginary parts of $\boldsymbol{y}^j$ are extracted and stored as a sample (size $2 \times N/N_t$) in the dataset. The cooperative training dataset also contains 20000 samples per SNR per modulation type. In both non-cooperative and cooperative cases, the input sequence length is chosen as 256 symbols. Hence, the size of each sample in the non-cooperative dataset is $8 \times 128$ for a $2 \times 4$ MIMO system and that for a cooperative dataset is $2 \times 128$.

**(c) Dataset for Training the ANN to Implement Feature-Fusion-Based Cooperative AMC**

In the case of feature-fusion-based AMC, an ANN is employed as the classifier. This ANN is trained with features extracted from the final dense layer of the CNN network. A dataset consisting of the feature vectors obtained at the final hidden layer of the modified DDrCNN is generated by using the CNN training data.

### 4.4.2 Performance evaluated on non-cooperative AMC dataset

In the non-cooperative scenario, the signals at the receiving antenna are fed to the CNN as a single frame of data (IQ format). In this case, each of the samples in the dataset is of size $(2N_r \times N/N_t)$, where N is the number of modulated symbols. The MIMO configuration $2 \times 4$ is selected and the input symbol size considered is 256. Hence, for the non-cooperative case, the input to the CNN will be $8 \times 128$ data frame.

DDrCNN, DBCNN and the modified DDrCNN were implemented using the Keras framework on Nvidia GeForce GTX 1660 SUPER. For the comparison of the classification performance of different CNN architectures, a dataset that contains 5000 samples per SNR per modulation type for training and 2000 samples per SNR per modulation type each for validation and testing is generated. The percentage of correct classification (PCC) is considered the metric to

TH-4097_156102033

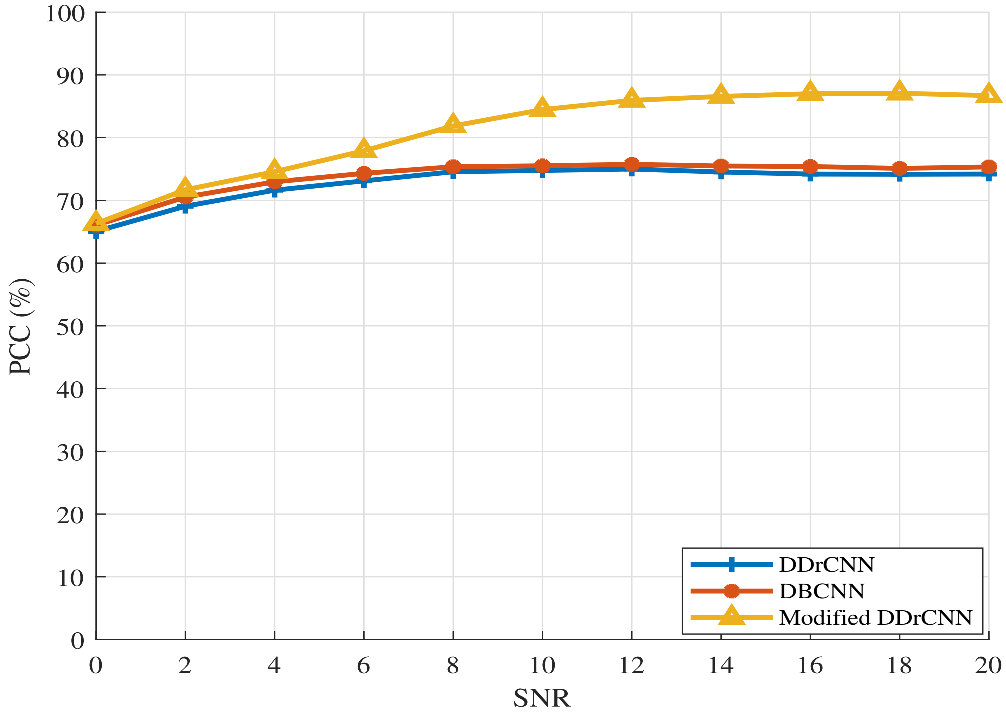



Figure 4.5: Non-cooperative DL-AMC of MIMO signals over Rayleigh fading channel ($N_t$ = 2, $N_r$ = 4).

evaluate the classifier's effectiveness. The networks were trained for 100 epochs, and an early stopping callback of patience 20 was used. Adam, the adaptive learning rate optimiser, is used to minimise the loss function. The learning rate value employed for the training is 0.001 and the training batch size selected is 400. After training, the test dataset is fed to the network, and the PCC is evaluated for each SNR.

The PCC for the three different classifiers are plotted in Fig. 4.5. The plot shows that the modified DDrCNN-based classifier's performance surpasses its predecessors by a significant margin of nearly 10% at higher SNRs. The performance improvement comes at the cost of an increase in the network size. The number of trainable parameters for the modified DDrCNN is around 6.2 million. Although the performance improved with the network modifications, the classification accuracy remained below 90%, even at high SNRs. This leads us to incorporate cooperative AMC methods for better classification in the case of MIMO signals. Since the classification performance is poor, the classifier is not tested under antenna correlation.

TH-4097_156102033

### 4.4.3 Results for cooperation among multiple antenna signals

The DL-AMC methods are evaluated for the cooperative cases, and the results are discussed in this Section. The performance of the classifier is tested under both uncorrelated and correlated channels.

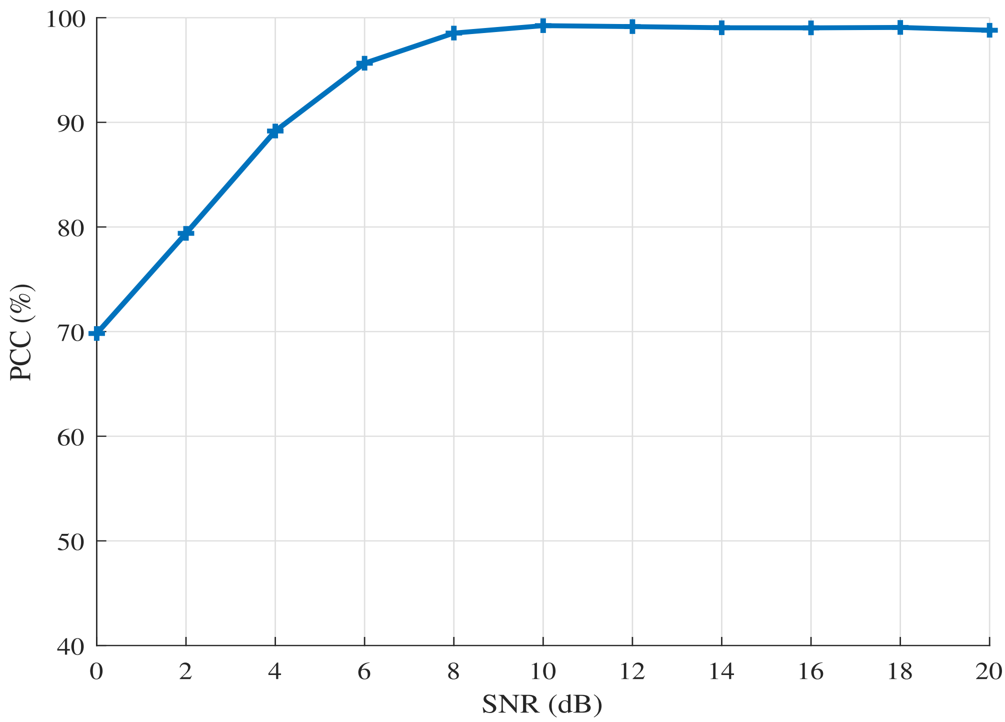



Figure 4.6: PCC for Modified DDrCNN-based CAMC (decision fusion) over MIMO Rayleigh channel for the 5-modulation pool ($N_t$ = 2, $N_r$ = 4)

TH-4097_156102033

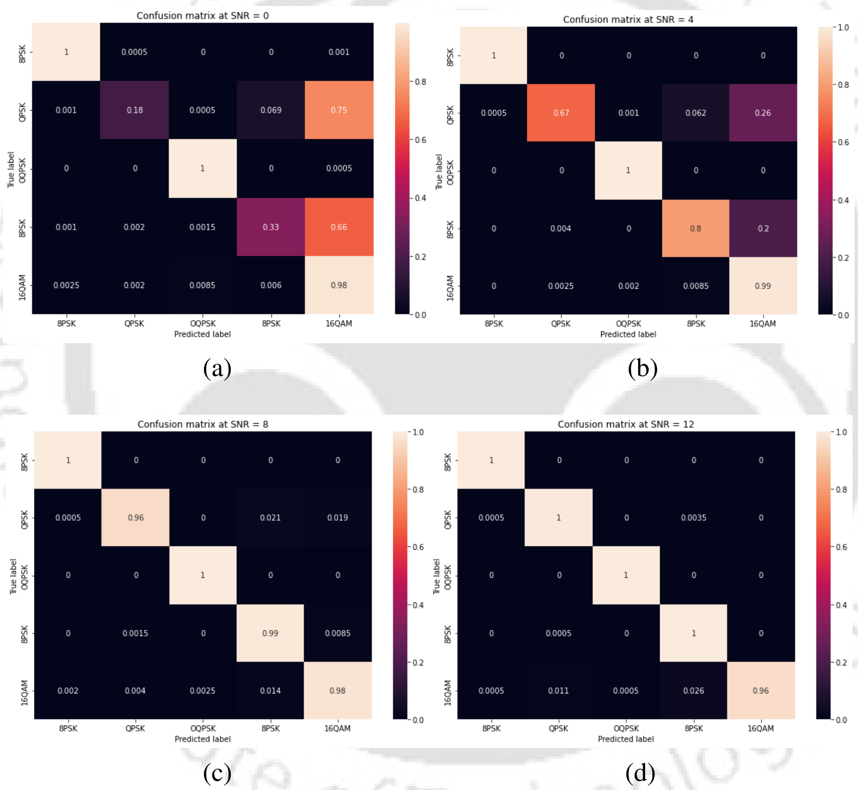



Figure 4.7: Confusion matrix of Modified DDrCNN at various SNRs on the MIMO dataset ($N_t$ = 2, $N_r$ = 4).

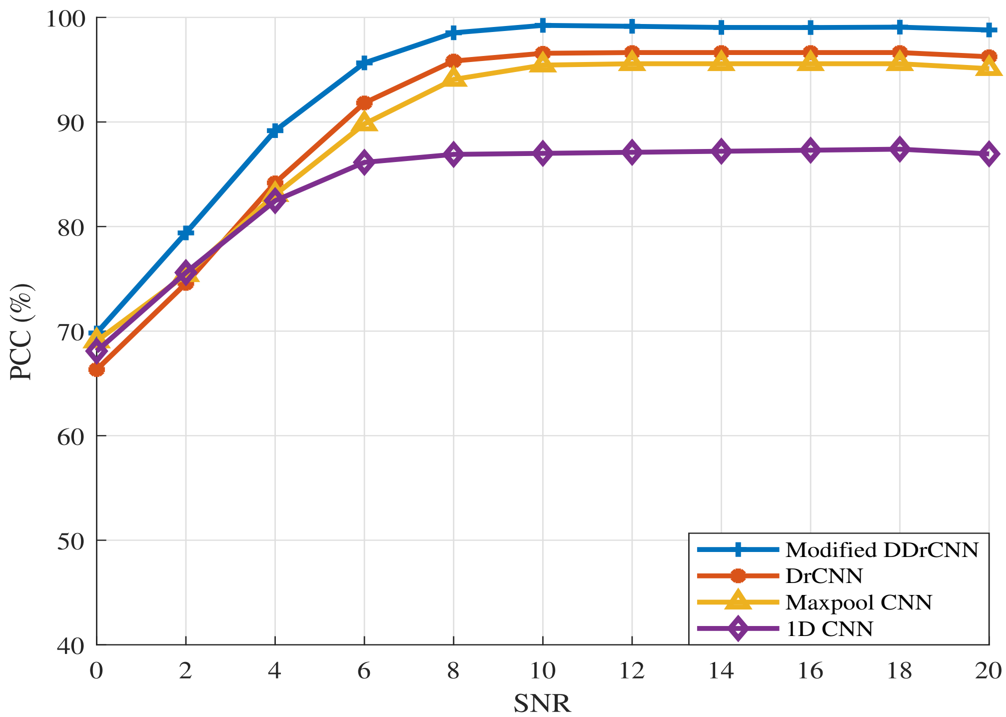



Figure 4.8: Comparison of PCC for DL-based CAMC (decision fusion) over MIMO Rayleigh channel for the 5-modulation pool ($N_t$ = 2, $N_r$ = 4)

**(a) Classification performance of modified DDrCNN :** The PCC for modified DDrCNN-based CAMC is plotted in Fig. 4.6. The confusion matrix of the modified DDRCNN classifier for a few SNRs is given in Fig. 4.7. At lower SNRs, the classifier detects QPSK and 8PSK as 16QAM, while BPSK and OQPSK are classified correctly even from 0 dB SNR.

**(a) Classification performance of modified DDrCNN and competing networks over uncorrelated MIMO channel:** The recognition performances of DL-classifiers with receiver cooperation for the 5-modulation pool are presented in Fig. 4.8. From the figures, one can observe that there is considerable improvement in performance. Modified DDrCNN was able to achieve more than 99% accuracy above 10 dB SNR. The 1D-CNN had the poorest performance among all networks. Table 4.2 compares the number of tunable parameters and the training time per epoch for the network under consideration. In terms of the number of parameters in the network, Maxpooling CNN is the lightest among all followed by modified DDrCNN. Consequently the training time also follows the same pattern.

TH-4097_156102033

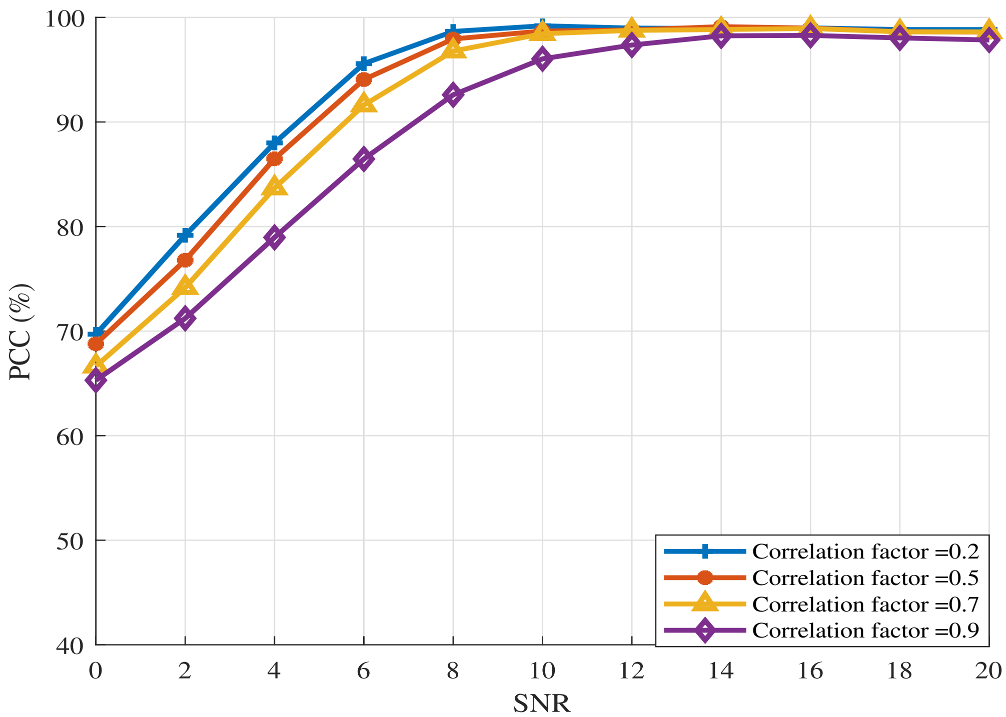



Figure 4.9: PCC for cooperative AMC over correlated MIMO channel ($N_t = 2$, $N_r = 4$)

**(b) Cooperative AMC under correlated MIMO channel:** The performance of DL-based AMC classifiers over correlated MIMO channels is elaborated in this experiment. The classification results for the 5-modulation pool under CAMC are presented. For testing the classifier under various correlation factors, datasets with different antenna correlations ($\rho = 0.2, 0.5, 0.7, 0.9$) at the transmitter and receiver are generated according to the model in equation (4.2) and (4.3). These datasets were tested using the modified DDrCNN-based classifier trained with uncorrelated data, and the result is plotted in Fig. 4.9. It can be observed that even in correlated channels, the proposed classifier is able to achieve decent classification performance. The network was able to retain an accuracy of above 96% for SNR 10 dB for the worst correlated

Table 4.2: Comparison of the number of trainable parameters and training time for the networks under consideration.

| CNN employed | Trainable parameters | Time per epoch |
|---|---|---|
| Maxpooling CNN | 3,30,372 | 45 seconds |
| DrCNN | 10,34,756 | 90 seconds |
| 1D CNN | 35,77,156 | 132 seconds |
| Modified DDrCNN | 9,96,836 | 70 seconds |

TH-4097_156102033

channel.

### 4.4.4 Classification results for classifier employing feature fusion.

The classification performance of feature fusion-based cooperation for AMC is presented here. The classification performance for the 2x4 MIMO classifier is plotted in Fig. 4.10. The result shows that, at lower SNRs, the feature fusion-based cooperation helped achieve better classification accuracy. The improvement in accuracy in lower SNRs may be attributed to the rich representation of data in the case of feature fusion. Combining features from multiple antennas improves performance because different antennas experience independent fadings. When combining the features, the classifier gets a good version of the waveform with higher effective SNR. At high SNRs, both decision fusion and feature fusion deliver similar performance.

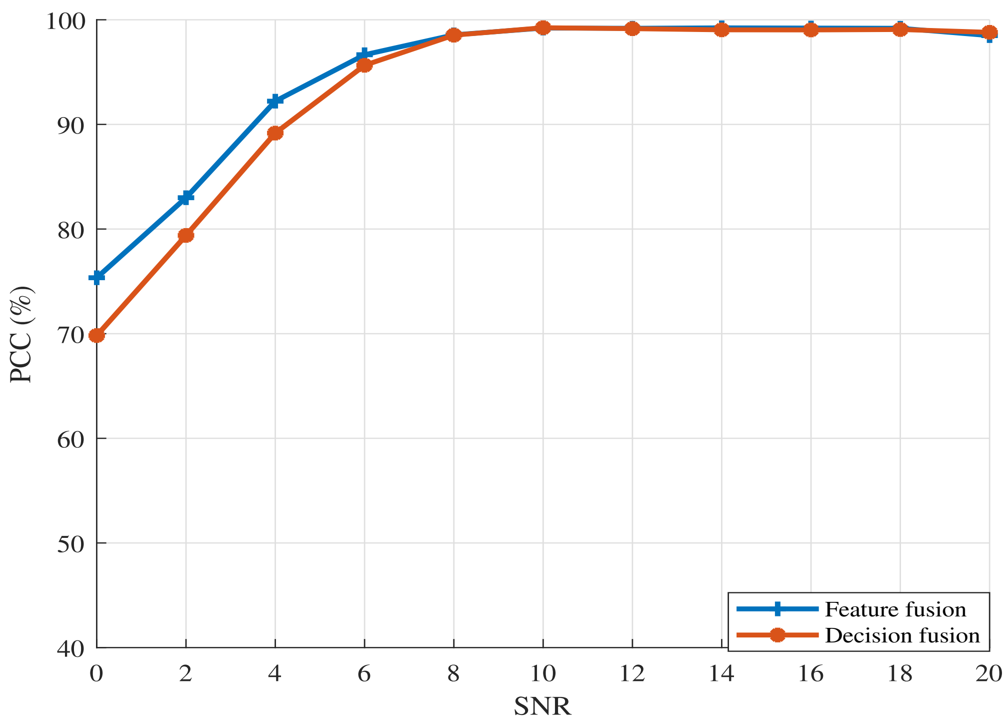



Figure 4.10: PCC for decision fusion vs feature fusion over uncorrelated MIMO channel ($N_t$ = 2, $N_r$ = 4) .

TH-4097_156102033

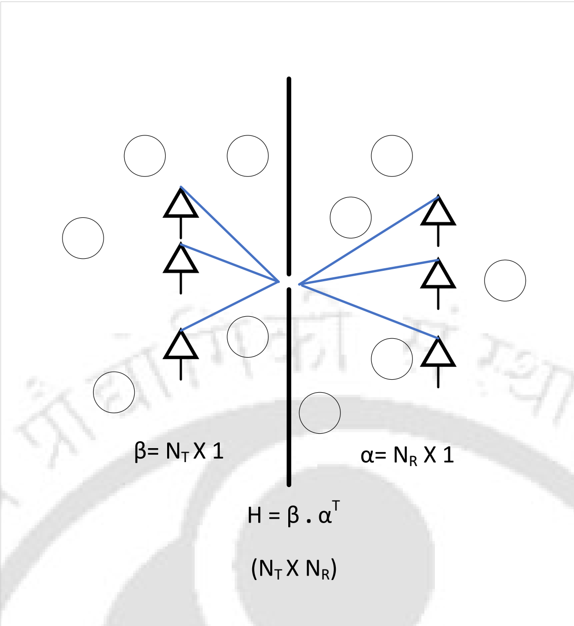



Figure 4.11: Representation of a spatial keyhole channel.

## 4.5 AMC Performance of the Modified DDrCNN on MIMO Signals Over Keyhole Channel.

In this section, the modified DDrCNN-based classifier is evaluated for its accuracy over rank-deficient MIMO channels. Specifically, the keyhole channel model, which represents the extreme case of rank deficiency, is considered. The section begins with an explanation of the keyhole channel model.

### 4.5.1 MIMO Keyhole Channel

The MIMO model given in equation 4.1 represents a full-rank channel. A rank-deficient channel refers to a channel whose channel matrix, H, has a rank less than the minimum number of transmit and receive antennas in a MIMO communication system. The rank of the MIMO channel matrix represents the number of independent spatial paths present in the channel. The spatial diversity of the MIMO system is compromised in the rank-deficient scenario.

TH-4097_156102033

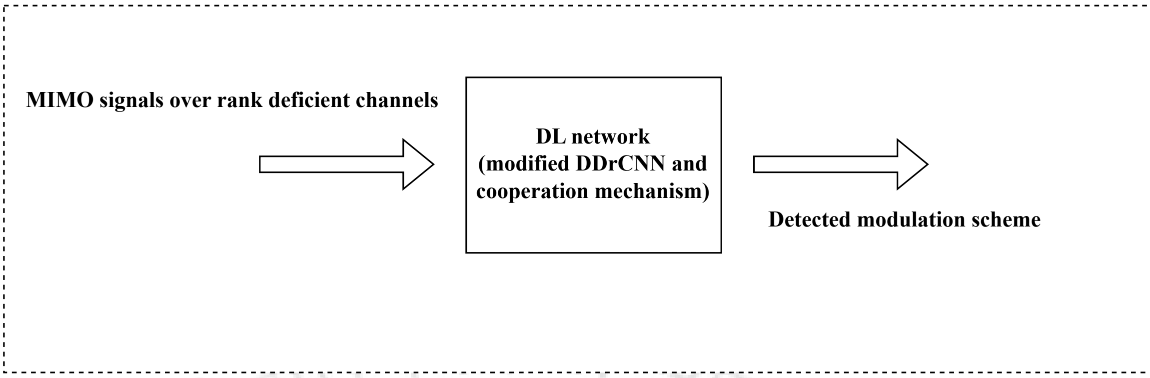



Figure 4.12: CNN classifier for signals over rank deficient channels.

For instance, consider a scenario where an indoor user device transmits signals to a cell tower. The signal transmitted by the device may predominantly pass through openings like windows and doors due to the signal-absorbing nature of building walls. Such occurrences of signal passage through keyholes are also possible outdoors, even in environments with significant scattering. A keyhole channel, an example of a rank-deficient channel, can occur because of a rich scattering environment separated by a large distance or rich scattering environments connected by a rank-1 propagation path. Rich scattering environments connected by diffraction over edges also create a keyhole channel [57, 59]. This can be explained using Fig 4.11, where small keyhole is punched through a screen, separating the transmitter and receiver antennas. The model [63] that represents a MIMO-correlated keyhole channel vector can be expressed as,

$$\boldsymbol{H} = \boldsymbol{\beta}\boldsymbol{\alpha}^{T} \tag{4.5}$$

where $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ are independent Rayleigh vectors of size $N_R \times 1$ and $N_T \times 1$ respectively. $\boldsymbol{\beta}$ represents the channel vector from the transmitter to the keyhole, and $\boldsymbol{\alpha}$ represents the channel from the keyhole to the receiver.

### 4.5.2 Various Training Scenarios Explored for the Modified DDrCNN

In the previous Section, the CNN-based classifier for the MIMO Rayleigh channel was explored. This Section investigates the suitability of the same AMC classifier for signals over keyhole channels as in Fig. 4.12. For this study, the decision fusion-based cooperation is considered. Two more datasets were generated to train the classifier for the evaluation. The

TH-4097_156102033

following sub-section details the different scenarios considered.

**Case 1: MIMO keyhole signals tested on CNN-classifier trained on similar training data**

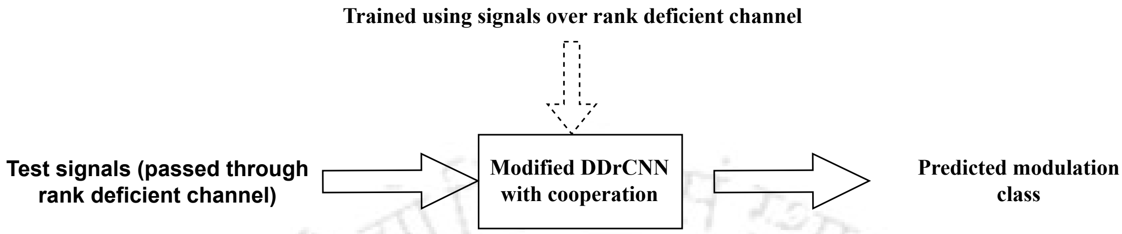



Figure 4.13: AMC employing the SCNN classifier trained with signals over keyhole channel

In the first investigation, the performance of the modified DDrCNN trained on signals over rank-deficient channels is tested. Fig. 4.13 represents the process. A train dataset and a test dataset consisting of signals passed through a keyhole channel are generated utilising the channel matrix given in 4.5. The dataset consist of five modulation schemes, viz. BPSK, QPSK, OQPSK, 8-PSK and 16-QAM. Initially, a random sequence of length N is generated, which is then modulated to produce the complex modulated vector **X** of the same size. The vector **X** is normalised to unit power and reshaped into $N_t \times N/N_t$, where $N_t$ is the number of transmitting antennas of the MIMO system. The signal is then passed through the keyhole channel and received at each of the receiving antennas. The real part and imaginary part of the received signal are extracted and stored as a sample of size $2 \times N/N_t$ in the dataset.

**Case 2: MIMO keyhole signals tested on CNN-classifier trained on full rank MIMO signals**

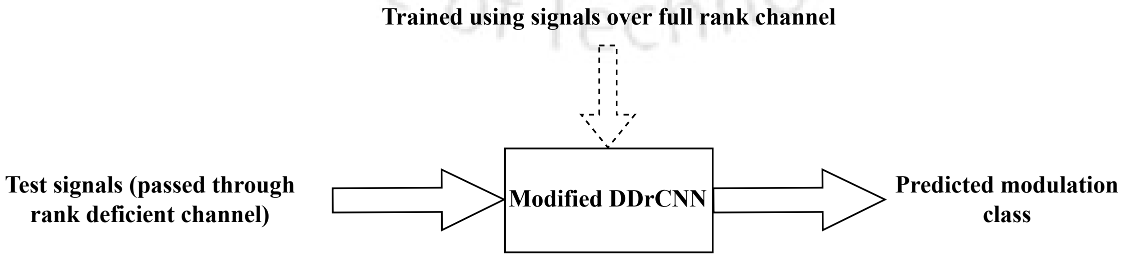



Figure 4.14: AMC employing the SCNN classifier trained with signals over a full rank channel.

In the second case, the modified DDrCNN network trained on the MIMO signals over Rayleigh channel is utilised as the classifier, as illustrated in Fig.4.14. The signals passed through the keyhole channel are tested, and the performance is evaluated.

TH-4097_156102033

**Case 3:- MIMO keyhole signals tested on CNN-classifier trained on the mixed dataset**

In this scenario, a dataset consisting of signals passed through the full-rank channels as well as the rank-deficient channels is created. A 50:50 ratio is used while generating the dataset. Hence, in the dataset for each SNR and each selected modulation, if the first sample is from the full rank channel, the second is from the rank-deficient channel. The modified DDrCNN is trained using this new dataset, keeping all the other training parameters the same. The resulting network is employed as the classifier for the signals over the rank-deficient channel.

### 4.5.3 Results

This Section discusses the results of the AMC using a modified DDrCNN-based classifier for the MIMO signals over rank-deficient channels. For each case, the CNN was trained using the appropriate dataset, the performance was evaluated, and the PCC was plotted. In general, the training data set contains 5000 samples, and the test data set contains 2000 samples per SNR per modulation type.

**(a) CAMC performance of modified DDrCNN-based classifier trained on signals over a rank-deficient channel:** The classifier was trained on the dataset containing signals over the keyhole channel. While training the model, a batch size of 400 is selected and an early stopping callback of patience 10 is used to combat over-fitting. The loss function selected was categorical cross-entropy, and the optimiser was Adam, similar to the previous works.

For testing the classifier, a set of signals whose SNR ranges from 0 to 20 dB is used. The corresponding PCC is shown in Fig. 4.15. It is observed that the PCC increases from 63% at 0 dB to 97% at 20 dB.

TH-4097_156102033

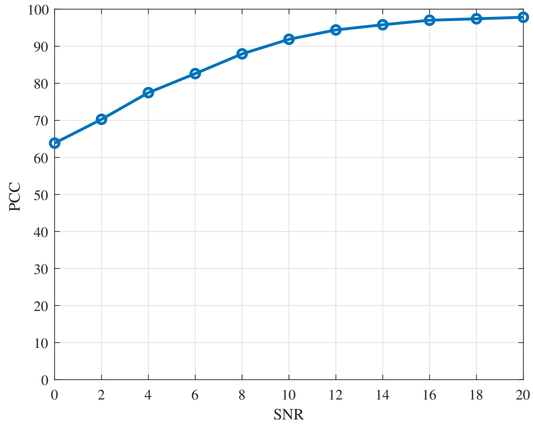



Figure 4.15: PCC for keyhole channels while the classifier was trained with similar signals

**(b) AMC performance of modified DDrCNN-based classifier trained on signals over full rank channel:** The test dataset containing signals over a rank 1 channel is fed to the modified DDrCNN-based classifier trained using the MIMO Rayleigh signals. The PCC for this scenario is plotted in Fig. 4.16. From the plot, the PCC performance appears to be slightly better than in the previous case.

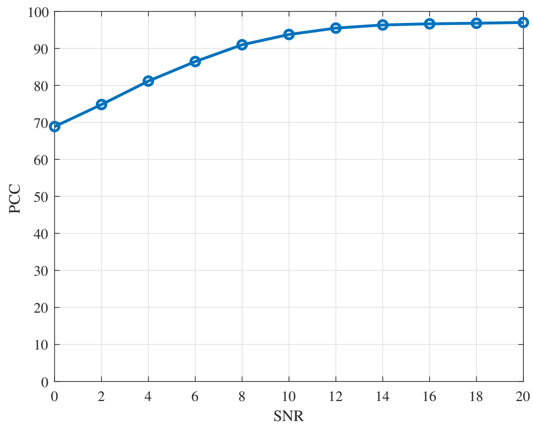



Figure 4.16: PCC for keyhole channels while the classifier is trained with signals over a full-rank channel

**(c) AMC performance of modified DDrCNN-based classifier trained on a mixed signals dataset:** In this scenario, the test signals are evaluated using a classifier trained on a mixed dataset comprising signals transmitted through both rank-deficient and full-rank channels. The classifier's training process followed the same methodology as in the first case, with all training

TH-4097_156102033

parameters remaining consistent. The PCC of the classifier for the test dataset is shown in Fig. 4.17. The performance in this scenario was slightly lower compared to the previous two cases.

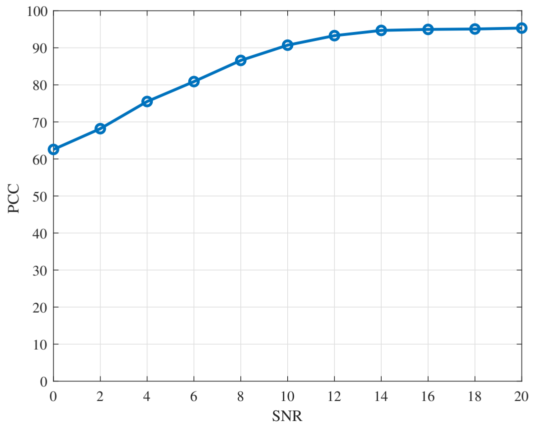



Figure 4.17: PCC for the modified DDrCNN over keyhole channels where the classifier was trained with a mixed dataset

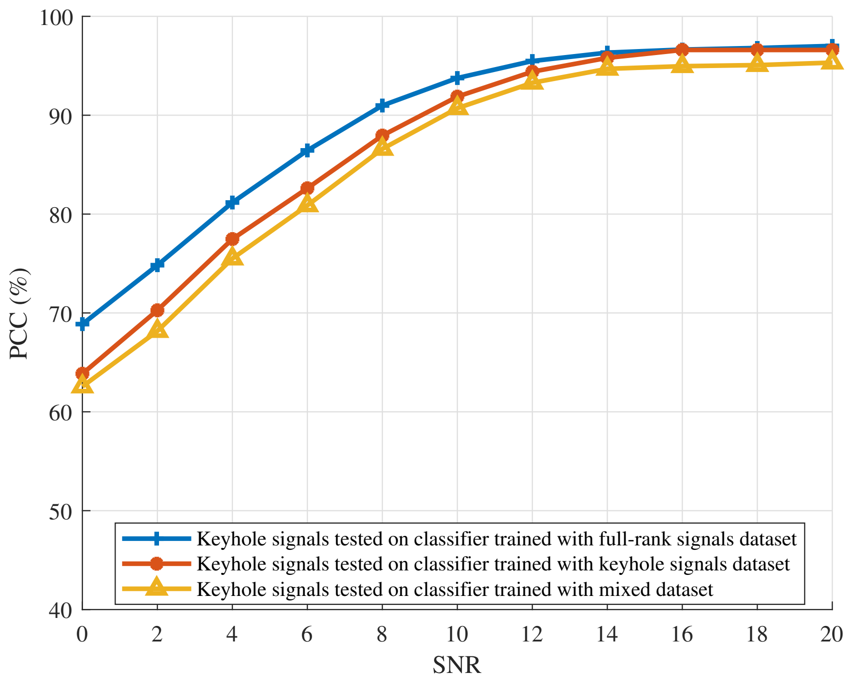



Figure 4.18: Comparison of the classification performance of modified DDrCNN trained on different datasets

From the PCC comparison plot given in Fig. 4.18, it is evident that case 1 and case 2 classification accuracies are similar from SNR 16 dB onwards, while around a 4% difference exists

TH-4097_156102033

at 0 dB. The classifier trained on the mixed dataset consistently lags behind in performance compared to both the other scenarios. Figure 4.19 pictures the confusion matrix at 0 dB, 10dB and 20 dB for the three scenarios. The confusion matrix shows that at high SNRs, cases 1 and 2 differ only in the detection of 16 QAM signals. Other modulation schemes return the same classification results. However, at lower SNRs, the accuracy of PSKs is poor for the classifier trained with signals over the rank-deficient channel and mixed dataset.

This behaviour is largely influenced by the type of signal variations seen during training. When the model is trained using signals passed through a full-rank channel, it gets exposed to richer spatial and temporal diversity, which helps it learn more discriminative and channel-invariant features. Because of this, the classifier trained on full-rank data is able to generalize better, even if the test signals later come through a rank-deficient channel. On the other hand, when the training data contains only rank-deficient or mixed-channel signals, the model sees a much narrower and more distorted feature space, leading it to overfit to those restricted conditions. As a result, these models struggle when the SNR is low or when the signal characteristics vary beyond what they observed during training.

TH-4097_156102033

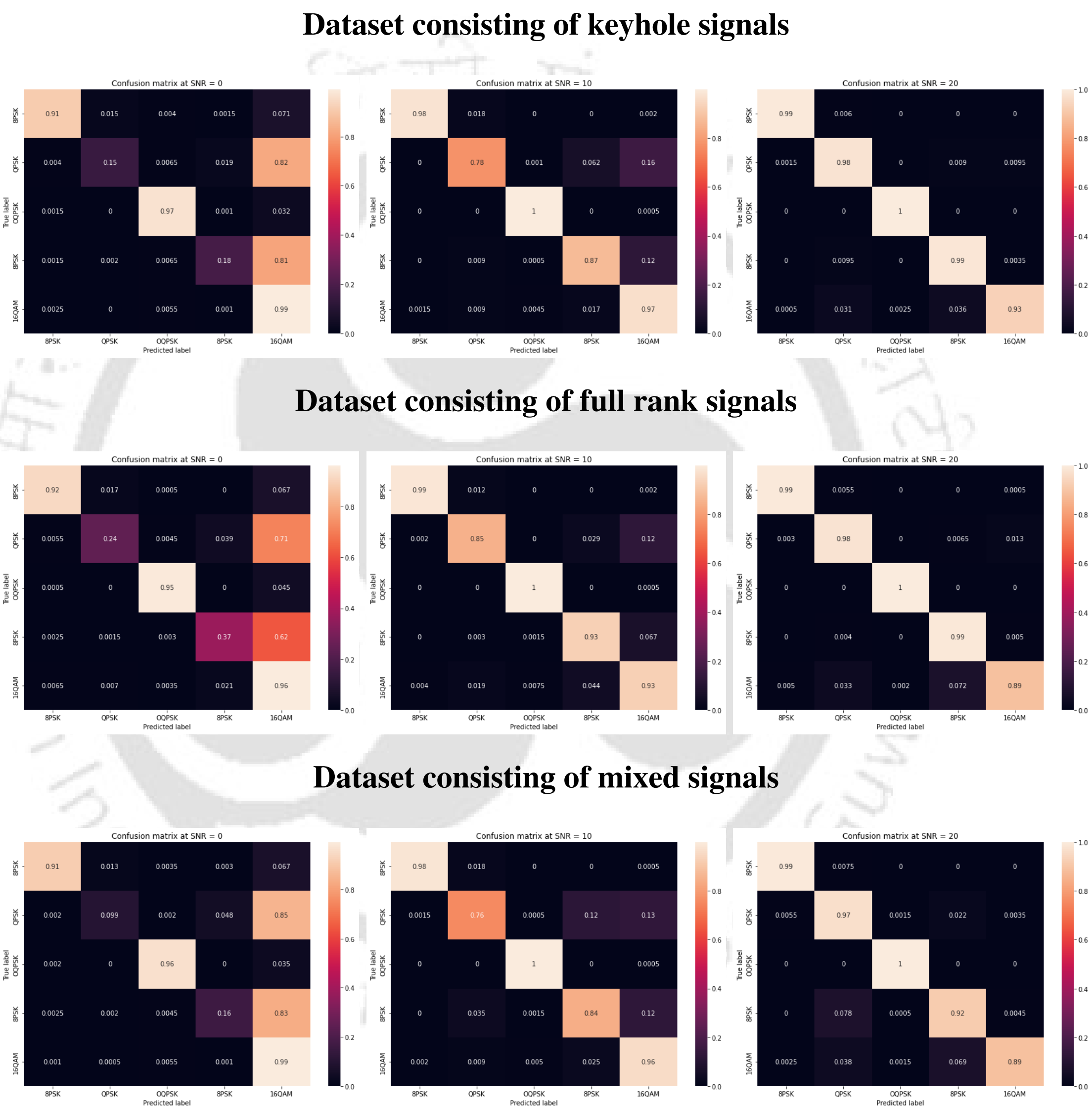



Figure 4.19: Confusion matrices showing the classification performance of the modified DDrCNN on MIMO keyhole signals at different SNR levels, using models trained on separate datasets

#### 4.5.3.1 Comparison of SCNN-based AMC over rank-deficient channel with traditional FB-AMC method

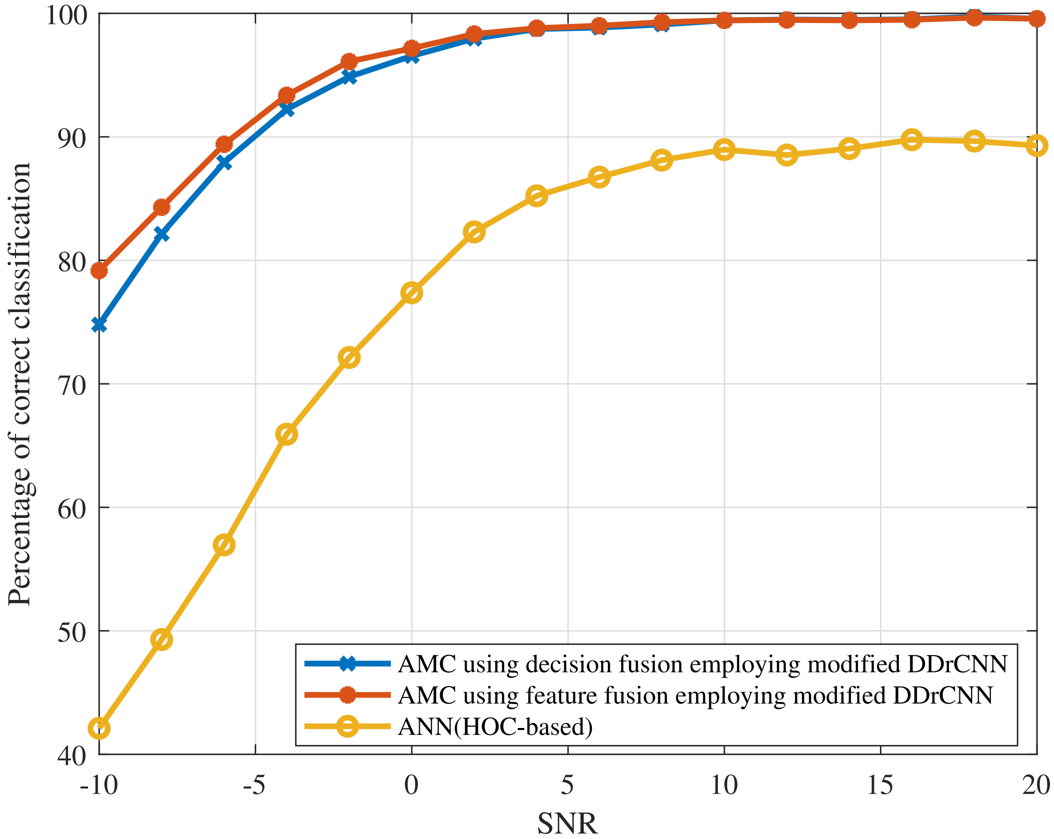



Figure 4.20: Comparison of AMC performance of modified DDrCNN-based classifier with the baseline HOC-based ANN AMC over MIMO keyhole channel for 3-modulation pool.

In [31], an HOC-based algorithm is proposed for the classification of lower-order PSK signals over the MIMO keyhole channel. The modulation pool considered was {BPSK, QPSK, OQPSK}. The direct modulation recognition (DMR) algorithm proposed in this work does not require CSI at the receiver for classification. The authors cleverly employed the ratio of $4^{th}$ and $6^{th}$ order cumulants to cancel out the channel effects. The discriminative features were ratios of HOCs as given by

$$\hat{f}_1 = \frac{\sum_{i=1}^{N_r} \hat{C}_{6,3}(\Delta y(k))}{\sum_{i=1}^{N_r} \hat{C}_{6,3}(y(k))} \tag{4.6}$$

and

$$\hat{f}_2 = \frac{\sum_{i=1}^{N_r} \hat{C}_{4,2}(\Delta y(k))}{\sum_{i=1}^{N_r} \hat{C}_{4,2}(y(k))} \tag{4.7}$$

where $\Delta\hat{y}(k) = \hat{y}(k) - y(k-1)$ is the backward difference of the received signal and $\hat{C}_{m,n}$is the estimate of the $m^{th}$ order cumulant value of signal. For an in-depth understanding of these features, one can refer to [31].

Since the HOC-based AMC of MIMO signals over a keyhole channel was proposed to classify lower-order PSKs, the classification result of modified DDrCNN for the same modulation pool is used for the comparison. For the training of the ANN using the features mentioned in the equations (4.6) and (4.7), a dataset consisting of 10000 feature vectors per SNR per modulation type is created. In the baseline work, the authors considered 4000 samples at the receiver for classification purposes. However, for a fair comparison, only 128 samples were considered while generating the dataset. A minimum distance classifier is used in the baseline work to discriminate the extracted features, which is replaced by an ANN classifier in this comparison.

From the classification performance plotted in Fig. 4.20, it is easily observable that the CNN-based classifier outperforms the HOC-based ANN classifier employing the DMR technique in a MIMO keyhole channel. The proposed CNN-based method achieved 99% classification above 5 dB SNR while the baseline method was only able to achieve a maximum of 90% classification at around 10 dB SNR.

## 4.6 Summary

This chapter presented the investigation on the performance of the SCNN-based AMC over a MIMO Rayleigh channel. A modified DDrCNN-based classifier is proposed. This classifier was able to outperform the networks presented in the previous chapters. To improve the classification results, a cooperative decision fusion mechanism and a feature fusion mechanism were employed at the receiver. At lower SNRs, the feature fusion-based classifier returned better performance. The performance of a few similar classifiers presented in the literature was inves-

TH-4097_156102033

tigated using the MIMO data set. The study found that the modified DDrCNN-based classifier outperformed its peers. The AMC performance of MIMO signals over a keyhole channel is investigated on an SCNN-based classifier trained using different datasets. The classifier trained on the signals through a full-rank channel appears to perform better than others. The classifier trained on full rank signals attained a maximum accuracy of 96.5 % at 14 dB, while that trained on keyhole signals attained similar classification at around 16 dB. A study is carried out to compare the performance of SCNN-based AMC with the FB-based AMC for rank-deficient channels. In comparison with the baseline HOC-based method, the proposed DL-based method achieved better classification performance and was able to classify the higher-order constellations like 8-PSK and 16-QAM.

TH-4097_156102033

# 5

# Conclusions and Future Works

TH-4097_156102033

## 5.1 Conclusions

Automatic modulation classification employing DL techniques is currently in the early stages of research. A number of research studies on the DL-AMC of both SISO and MIMO signals were reported. This thesis carried out a detailed analysis of existing DL-AMC methods using SCNNs and proposed modifications to improve classification performance. The contributions from this study were presented in three chapters, which are summarised below.

**Chapter 2** explored the AMC of 8 digital modulation classes, namely BPSK, QPSK, 8PSK, GFSK, CPFSK, PAM4, 16QAM, and 64QAM. The well-known classifier based on DrCNN employed a dual CNN-based classifier to discriminate between these modulation schemes. The research work in this chapter established that eliminating dropouts from the convolutional layers enhanced the classification accuracy. The DDrCNN and DBCNN proposed in this thesis could effectively classify the eight modulation schemes while significantly reducing the network size. The DDrCNN and DBCNN models were designed to be parameter-efficient, containing approximately 21% of the trainable parameters of the original DrCNN architecture. The study on the effect of the size of the received signal on the classification showed that 128 samples of the incoming signal are a good selection for DL-based classifiers. The proposed classifier was also evaluated on the publicly available dataset RadioML2018.01.

Researchers have observed that cooperation among multiple receivers enhances AMC performance. Since many communication systems employ multiple nodes at the receiver, **Chapter 3** of this thesis investigated the performance improvement achievable through decision-level cooperation of multiple sensor nodes employing SCNN classifiers. Most of the cooperative AMC systems have utilised cumulant-based features, which require a larger number of received signal samples for classification, and whose ability to discriminate between modulation classes depends heavily on the selected feature set, thereby limiting the range of modulation schemes that can be supported. In this work, a centralised cooperation architecture was adopted, in which a fusion centre aggregates the decisions generated by CNN classifiers at individual receiving nodes to produce the final classification result. For an input size of 128 samples, averaging-based cooperation improved the prediction accuracy at $-10$, dB SNR from 89.5% for a single

TH-4097_156102033

node to 99.5% when five cooperating nodes were employed, with the maximum achieved PCC reaching 100% at SNRs of $-4$, dB and above. For an input size of 64 samples, the same cooperation strategy improved the prediction accuracy at $-10$,dB SNR from 69.66% for a single node to 84.31% with 10 cooperating nodes, with the maximum achieved PCC reaching 98% at 6 dB SNR. These results demonstrate that cooperation significantly enhances AMC performance even with reduced input sample lengths, thereby making it suitable for hardware constraint applications.

A modified version of DDrCNN was proposed for the AMC of MIMO signals in Chapter 4. The modified architecture included three convolutional layers and two dense layers having 256 and 128 neurons, respectively. For evaluating the AMC performance, the $2\times4$ MIMO configuration is selected. A decision cooperation mechanism at the MIMO receiver is employed to enhance the PCC. The performance of the modified DDrCNN is compared with that of the competing methods. On the generated dataset containing modulation classes, BPSK, QPSK, OQPSK, 8-PSK, and 16-QAM, the modified DDrCNN was able to achieve more than 99% accuracy at above 10 dB SNR. The classifier was also tested under a correlated MIMO channel and found to be performing satisfactorily. In addition to the averaging-based decision cooperation, a feature fusion-based cooperation method was also evaluated. The feature fusion method employed a concatenated feature vector from the final dense layer. The feature fusion cooperation was found to marginally outperform the decision fusion technique at lower SNRs. But if considering the number of parameters and efficiency decision fusion is preferred.

In a typical communication system, the MIMO channel is assumed to have a rich scattering environment. In practical situations, the channel may have insufficient scattering, leading to the existence of reduced-rank MIMO channels. It is reported in the literature that the HOC-based FB methods fail to classify the modulation schemes if low-rank channels exist. In Chapter 4, the operation of CNN-based AMC on keyhole channels—an extreme case of rank-deficient channels—was also investigated. The CNN-based classifier was trained on multiple data sets and found that the network trained on the full-rank channel data was able to classify the signals passed through the keyhole channel. The classifier achieved over 99% accuracy at 10 dB SNR

TH-4097_156102033

for a modulation pool consisting of BPSK, QPSK, OQPSK, 8-PSK and 16-QAM. The PCC for the classifier trained on a mixed data set containing signals from the full-rank channel and a keyhole channel (50% – 50%) marginally underperformed with around 98% accuracy at 10 dB, but it underperformed by a wide margin at lower SNRs. The performance of the modified DDrCNN on a dataset containing BPSK, QPSK and OQPSK was evaluated against the baseline FB method employing the DMR algorithm for the keyhole channel. For signals of sample size 128, the proposed classifier achieved 99% accuracy at 5dB while the baseline method achieved a maximum of 90% PCC at 10 dB.

## 5.2 Scope for Future Works

The work presented in this thesis addresses fundamental AMC problems under relatively simplified communication channel conditions. Although the shallow CNN model developed in this thesis performs effectively for the considered set of modulation classes, the same architecture may not scale well when the number of classes increases substantially. Future research may therefore explore alternative shallow structures or enhanced feature representations that preserve low complexity while enabling reliable AMC performance across a larger and more diverse modulation pool. To move towards a comprehensive and deployable AMC system leveraging DL models, several research directions remain open for exploration.

Notably, the majority of the datasets used in the current literature are synthetically generated under controlled conditions. The reliability of these datasets is still being widely discussed by active researchers [64, 65]. To enhance the robustness and generalizability of AMC systems, it is essential to develop and utilise more realistic datasets comprising real-time signals captured across diverse and dynamically varying channel environments. Such practical datasets will be critical in designing AMC frameworks that are resilient, reliable, and capable of operating effectively in real-world deployment scenarios.

Some of the important areas where research can be conducted are outlined below.

(i) **Evaluation with Practical Signals:** Despite the impressive performance of DL-based AMC models on synthetically generated datasets, there remains a critical gap in their real-

TH-4097_156102033

world applicability. Most current models are trained and validated using clean, simulated signals that lack real-world complexities such as hardware impairments, interference, and diverse channel dynamics. For DL-AMC to transition from research to deployment, it is essential to evaluate and fine-tune these models using practical signals. The evaluations may include:

- Over-the-Air (OTA) signal recordings, which account for realistic impairments and variations due to hardware, multipath propagation, Doppler shifts, and interference.
- Diverse channel and interference conditions, such as time-varying fading, mobility-induced distortions, and multi-user interference, to assess robustness and generalisation.

(ii) **Evaluation under Standardised 3GPP Channel Models:** The channel models adopted in this thesis—the Rician, Rayleigh, and keyhole models—are analytically tractable baselines that isolate the effect of small-scale fading, but they do not reflect the full complexity of the channel models adopted in the industry. A natural extension of this work is the evaluation and retraining of the proposed shallow CNN classifiers under standardised 3GPP channel models, such as the Clustered Delay Line (CDL) and Tapped Delay Line (TDL) profiles specified in 3GPP TR 38.901, which additionally incorporate frequency-selective multipath, power-delay profiles, mobility-induced Doppler spectra, and spatial correlation. Such an evaluation would establish the practical relevance of the proposed classifiers for deployment in realistic, industry-compliant scenarios.

(iii) **Enhanced Feature Representation:** The existing deep learning models such as CNNs and RNNs have shown effectiveness in learning modulation-specific features from raw IQ samples or constellation diagrams, however performance degradation is still observed under time-varying or highly noisy channels. Fusion of domain-specific features such as cyclic spectrum, eye diagrams, and time-frequency transforms with learned representations may be investigated. Conventional CNNs operate on Euclidean grids (e.g., 2D images or 1D sequences), assuming fixed spatial locality. However, communication sig-

TH-4097_156102033

nals often exhibit non-Euclidean relationships — for instance, correlations across different frequency bins or among subcarriers in OFDM systems. In a graph-based AMC framework, each node can represent a signal component (e.g., a time frame, subcarrier, or frequency bin), while edges encode relationships such as correlation, coherence, or mutual information between components. The Graph Neural Networks (GNNs) then propagates and aggregates information across these edges, allowing the network to learn global structural dependencies beyond fixed kernel neighborhoods [66, 67]. Future research can focus on developing hybrid AMC frameworks that fuse domain-specific signal representations with graph-based neural architectures, enabling the model to capture both local and global dependencies and thereby improve classification performance in dynamic and non-stationary wireless environments.

(iv) **Classification Under Time-Varying Channels:** Traditional deep learning classifiers such as CNNs and feedforward networks treat IQ samples as static inputs, thereby neglecting the temporal dynamics inherent in wireless signal propagation. However, in realistic wireless environments, signal characteristics—such as amplitude, phase, and frequency—continuously evolve over time due to factors like Doppler shifts, mobility, and channel fading. Temporal modelling aims to explicitly capture these time-dependent variations to enhance classification robustness under non-stationary channel conditions.

Hierarchical Recurrent Neural Networks can be employed to model short- and long-term dependencies within the signal sequence. These networks effectively learn temporal correlations and can adapt to gradual or abrupt changes in channel state information. Alternatively, Attention-Based Transformers provide a more flexible mechanism by learning contextual dependencies without relying on sequential recurrence, allowing them to focus selectively on the most informative time segments. Such transformer-based models can efficiently capture long-range temporal relationships and are particularly advantageous in handling complex, time-varying fading patterns.

Future research may therefore explore hybrid temporal-spatial architectures that combine

TH-4097_156102033

convolutional feature extraction with attention or recurrent mechanisms to achieve improved adaptability and performance across dynamic channel conditions.

Indian Institute of Technology Guwahati

TH-4097_156102033

# List of Publications

### Journal

- P. Dileep, A. Singla, D. Das and P. K. Bora, ”Deep Learning-Based Automatic Modulation Classification Over MIMO Keyhole Channels,” in IEEE Access, vol. 10, pp. 119566-119574, 2022.

### Conference

- P. Dileep, D. Das and P. K. Bora, ”Dense Layer Dropout Based CNN Architecture for Automatic Modulation Classification,” 2020 National Conference on Communications (NCC), Kharagpur, India, 2020.

TH-4097_156102033

# Bibliography


[1] O. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, “Survey of automatic modulation classification techniques: classical approaches and new trends,” *IET Commun.*, vol. 1, pp. 137–156, 2007.

[2] J. Sills, “Maximum-likelihood modulation classification for psk/qam,” in *MILCOM 1999. IEEE Military Communications. Conference Proceedings (Cat. No.99CH36341)*, vol. 1, 1999, pp. 217–220 vol.1.

[3] A. Polydoros and K. Kim, “On the detection and classification of quadrature digital modulations in broad-band noise,” *IEEE Transactions on Communications*, vol. 38, no. 8, pp. 1199–1211, 1990.

[4] C.-Y. Huan and A. Polydoros, “Likelihood methods for mpsk modulation classification,” *IEEE Transactions on Communications*, vol. 43, no. 2/3/4, pp. 1493–1504, 1995.

[5] O. Dobre, Y. Bar-Ness, and W. Su, “Higher-order cyclic cumulants for high order modulation classification,” in *IEEE Military Communications Conference, 2003. MILCOM 2003.*, vol. 1, 2003, pp. 112–117 Vol.1.

[6] C. Spooner, “Classification of co-channel communication signals using cyclic cumulants,” in *Conference Record of The Twenty-Ninth Asilomar Conference on Signals, Systems and Computers*, vol. 1, 1995, pp. 531–536 vol.1.

[7] M. Flohberger, W. Gappmair, and O. Koudelka, “Modulation classifier for signals used in satellite communications,” in *2010 5th Advanced Satellite Multimedia Systems Conference and the 11th Signal Processing for Space Communications Workshop*, 2010, pp. 198–202.


TH-4097_156102033

[8] A. Swami and B. Sadler, “Hierarchical digital modulation classification using cumulants,” *IEEE Transactions on Communications*, vol. 48, no. 3, pp. 416–429, 2000.

[9] H.-C. Wu, M. Saquib, and Z. Yun, “Novel automatic modulation classification using cumulant features for communications via multipath channels,” *IEEE Transactions on Wireless Communications*, vol. 7, no. 8, pp. 3098–3105, 2008.

[10] D.-C. Chang and P.-K. Shih, “Cumulants-based modulation classification technique in multipath fading channels,” *IET Communications*, vol. 9, no. 6, pp. 828–835, 2015.

[11] V. D. Orlic and M. L. Dukic, “Automatic modulation classification algorithm using higher-order cumulants under real-world channel conditions,” *IEEE Communications Letters*, vol. 13, no. 12, pp. 917–919, 2009.

[12] M. S. Muhlhaus, M. Oner, O. A. Dobre, H. U. Jokel, and F. K. Jondral, “Automatic modulation classification for mimo systems using fourth-order cumulants,” in *2012 IEEE Vehicular Technology Conference (VTC Fall)*, 2012.

[13] K. Hassan, I. Dayoub, W. Hamouda, C. N. Nzeza, and M. Berbineau, “Blind Digital Modulation Identification for Spatially-Correlated MIMO Systems,” *IEEE Transactions on Wireless Communications*, vol. 11, no. 2, pp. 683–693, 2012.

[14] D. Das, A. Anand, P. K. Bora, and R. Bhattacharjee, “Cumulant based automatic modulation classification of QPSK, OQPSK, Pi/4-QPSK and 8-PSK in MIMO environment,” in *2016 International Conference on Signal Processing and Communications (SPCOM)*, 2016.

[15] C. Spooner, “Classification of co-channel communication signals using cyclic cumulants,” in *Conference Record of The Twenty-Ninth Asilomar Conference on Signals, Systems and Computers*, vol. 1, 1995, pp. 531–536 vol.1.

TH-4097_156102033

[16] O. Dobre, Y. Bar-Ness, and W. Su, "Higher-order cyclic cumulants for high order modulation classification," in *IEEE Military Communications Conference, 2003. MILCOM 2003.*, vol. 1, 2003, pp. 112–117 Vol.1.

[17] O. A. Dobre, Y. Bar-Ness, and W. Su, "Robust qam modulation classification algorithm using cyclic cumulants," in *2004 IEEE Wireless Communications and Networking Conference (IEEE Cat. No. 04TH8733)*, vol. 2. IEEE, 2004, pp. 745–748.

[18] K. Kim, I. A. Akbar, K. K. Bae, J.-S. Um, C. M. Spooner, and J. H. Reed, "Cyclostationary approaches to signal detection and classification in cognitive radio," in *2007 2nd ieee international symposium on new frontiers in dynamic spectrum access networks*. IEEE, 2007, pp. 212–215.

[19] E. Karami and O. A. Dobre, "Identification of sm-ofdm and al-ofdm signals based on their second-order cyclostationarity," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 3, pp. 942–953, 2015.

[20] W. A. Jerjawi, Y. A. Eldemerdash, and O. A. Dobre, "Second-order cyclostationarity-based detection of lte sc-fdma signals for cognitive radio systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 3, pp. 823–833, 2015.

[21] M. Marey, O. Dobre, and R. Inkol, "Cyclostationarity-based blind classification of stbcs for cognitive radio systems," in *2012 IEEE International Conference on Communications (ICC)*, 2012, pp. 1715–1720.

[22] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Engineering Applications of Neural Networks*, C. Jayne and L. Iliadis, Eds. Cham: Springer International Publishing, 2016, pp. 213–226.

[23] T. O'Shea, "Gnu radio channel simulation," in *GNU Radio Conference*, vol. 2013, 2013.

TH-4097_156102033

[24] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, 2017.

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

[26] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Distributed Deep Learning Models for Wireless Signal Classification with Low-Cost Spectrum Sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, jul 2017.

[27] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, 2018.

[28] M. Abdelbar, W. H. Tranter, and T. Bose, "Cooperative cumulants-based modulation classification in distributed networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 446–461, 2018.

[29] Y. Wang, J. Gui, Y. Yin, J. Wang, J. Sun, G. Gui, H. Gacanin, H. Sari, and F. Adachi, "Automatic modulation classification for MIMO systems via deep learning and zero-forcing equalization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5688–5692, 2020.

[30] Y. Wang, J. Wang, W. Zhang, J. Yang, and G. Gui, "Deep learning-based cooperative automatic modulation classification method for MIMO systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4575–4579, 2020.

[31] D. Das, P. K. Bora, and R. Bhattacharjee, "Blind modulation recognition of the lower order PSK signals under the mimo keyhole channel," *IEEE Communications Letters*, vol. 22, no. 9, pp. 1834–1837, 2018.

TH-4097_156102033

[32] K. Tekbıyık, C. Keçeci, A. R. Ekti, A. Görçin, and G. K. Kurt, "Hisarmod: A new challenging modulated signals dataset," 2019. [Online]. Available: https://dx.doi.org/10.21227/8k12-2g70

[33] X. Ying, "An overview of overfitting and its solutions," in *Journal of physics: Conference series*, vol. 1168. IOP Publishing, 2019, p. 022022.

[34] E. Phaisangittisagul, "An analysis of the regularization between l2 and dropout in single hidden layer neural network," in *2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS)*, 2016, pp. 174–179.

[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, p. 1929 – 1958, 2014, cited by: 28274. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84904163933&partnerID=40&md5=b865fd654b3befc5d829dbe5d42b80c3

[36] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. pmlr, 2015, pp. 448–456.

[37] Y. Gal and Z. Ghahramani, "Bayesian convolutional neural networks with bernoulli approximate variational inference," *arXiv preprint arXiv:1506.02158*, 2015.

[38] H. Wu and X. Gu, "Towards dropout training for convolutional neural networks," *Neural Networks*, vol. 71, pp. 1–10, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608015001446

[39] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Communications*, vol. 14, no. 11, pp. 92–111, 2017.

TH-4097_156102033

[40] M. Kato, T. Koketsu Rodrigues, T. Abe, and T. Suganuma, "Exploiting radio frequency characteristics with a support unmanned aerial vehicle to improve wireless sensor location estimation accuracy," *IEEE Internet of Things Journal*, vol. 11, no. 24, pp. 39 570–39 578, 2024.

[41] F. Xu, Y. Zhu, F. Yang, X. Zhang, J. Mu, and H.-H. Chen, "Distributed modulation recognition for iot devices in data-limited applications," *IEEE Internet of Things Journal*, vol. 12, no. 14, pp. 26 740–26 752, 2025.

[42] G. B. Marković, "Cooperative modulation classification by using multiple sensors in dispersive fading channels," in *2014 22nd Telecommunications Forum Telfor (TELFOR)*, 2014, pp. 264–271.

[43] M. Abdelbar, B. Tranter, and T. Bose, "Cooperative modulation classification of multiple signals in cognitive radio networks," in *2014 IEEE International Conference on Communications (ICC)*, 2014, pp. 1483–1488.

[44] J. L. Xu, W. Su, and M. Zhou, "Distributed automatic modulation classification with multiple sensors," *IEEE Sensors Journal*, vol. 10, no. 11, pp. 1779–1785, 2010.

[45] J.-f. Chamberland and V. V. Veeravalli, "Wireless sensors in distributed detection applications," *IEEE Signal Processing Magazine*, vol. 24, no. 3, pp. 16–25, 2007.

[46] M. Abdelbar, B. Tranter, and T. Bose, "Cooperative cumulants-based modulation classification under flat rayleigh fading channels," in *2015 IEEE International Conference on Communications (ICC)*, 2015, pp. 7622–7627.

[47] G. B. Markovic and M. L. Dukic, "Cooperative amc schemes using cumulants with hard and soft decision fusion," in *2012 20th Telecommunications Forum (TELFOR)*, 2012, pp. 400–403.

[48] K. B. Letaief and W. Zhang, "Cooperative communications for cognitive radio networks," *Proceedings of the IEEE*, vol. 97, no. 5, pp. 878–893, 2009.

TH-4097_156102033

[49] G. Ganesan and Y. Li, "Cooperative spectrum sensing in cognitive radio, part i: Two user networks," *IEEE Transactions on Wireless Communications*, vol. 6, no. 6, pp. 2204–2213, 2007.

[50] J. L. Xu, W. Su, and M. Zhou, "Distributed automatic modulation classification with multiple sensors," *IEEE Sensors Journal*, vol. 10, no. 11, pp. 1779–1785, 2010.

[51] G. Markovic and M. Dukic, "Cooperative modulation classification with data fusion for multipath fading channels," *Electronics Letters*, vol. 49, no. 23, pp. 1494–1496, 2013. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/el.2013.1028

[52] X. Yan, X. Rao, Q. Wang, H.-C. Wu, Y. Zhang, and Y. Wu, "Novel cooperative automatic modulation classification using unmanned aerial vehicles," *IEEE Sensors Journal*, vol. 21, no. 24, pp. 28 107–28 117, 2021.

[53] V. Choqueuse, S. Azou, K. Yao, L. Collin, and G. Burel, "Blind modulation recognition for mimo communications," *IEEE Communication Conference*, 06 2008.

[54] K. Hassan, C. N. Nzeza, M. Berbineau, W. Hamouda, and I. Dayoub, "Blind modulation identification for mimo systems," in *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, 2010.

[55] M. Marey and O. A. Dobre, "Blind modulation classification algorithm for single and multiple-antenna systems over frequency-selective channels," *IEEE Signal Processing Letters*, vol. 21, no. 9, pp. 1098–1102, 2014.

[56] Z. Zhu and A. K. Nandi, "Blind modulation classification for MIMO systems using expectation maximization," in *2014 IEEE Military Communications Conference*, 2014, pp. 754–759.

[57] D. Chizhik, G. Foschini, M. Gans, and R. Valenzuela, "Keyholes, correlations and capacities of multi-element transmit and receive antennas," in *IEEE VTS 53rd Vehicular*

TH-4097_156102033

*Technology Conference, Spring 2001. Proceedings (Cat. No.01CH37202)*, vol. 1, 2001, pp. 284–287 vol.1.

[58] S. Loyka and A. Kouki, “On MIMO channel capacity, correlations, and keyholes: analysis of degenerate channels,” *IEEE Transactions on Communications*, vol. 50, no. 12, pp. 1886–1888, 2002.

[59] P. Almers, F. Tufvesson, and A. F. Molisch, “Keyhole effect in MIMO wireless channels: Measurements and theory,” *IEEE Transactions on Wireless Communications*, vol. 5, no. 12, pp. 3596–3604, 2006.

[60] D.-S. Shiu, G. Foschini, M. Gans, and J. Kahn, “Fading correlation and its effect on the capacity of multielement antenna systems,” *IEEE Transactions on Communications*, vol. 48, no. 3, pp. 502–513, 2000.

[61] K. Hassan, I. Dayoub, W. Hamouda, C. N. Nzeza, and M. Berbineau, “Blind digital modulation identification for spatially-correlated MIMO systems,” *IEEE Transactions on Wireless Communications*, vol. 11, no. 2, pp. 683–693, 2012.

[62] P. Dileep, D. Das, and P. K. Bora, “Dense layer dropout based CNN architecture for automatic modulation classification,” in *2020 National Conference on Communications (NCC)*, 2020.

[63] J. Salo, H. El-Sallabi, and P. Vainikainen, “Impact of double-rayleigh fading on system performance,” in *2006 1st International Symposium on Wireless Pervasive Computing*, 2006.

[64] V. Sathyanarayanan, P. Gerstoft, and A. E. Gamal, “Rml22: Realistic dataset generation for wireless modulation classification,” *IEEE Transactions on Wireless Communications*, vol. 22, no. 11, pp. 7663–7675, 2023.

[65] J. A. Snoap, J. A. Latshaw, D. C. Popescu, and C. M. Spooner, “Robust classification of digitally modulated signals using capsule networks and cyclic cumulant features,” in

TH-4097_156102033

*MILCOM 2022 - 2022 IEEE Military Communications Conference (MILCOM)*, 2022, pp. 298–303.

[66] A. G. Vrahatis, K. Lazaros, and S. Kotsiantis, "Graph attention networks: A comprehensive review of methods and applications," *Future Internet*, vol. 16, no. 9, 2024. [Online]. Available: https://www.mdpi.com/1999-5903/16/9/318

[67] D. Wang, M. Lin, X. Zhang, Y. Huang, and Y. Zhu, "Automatic modulation classification based on cnn-transformer graph neural network," *Sensors*, vol. 23, no. 16, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/16/7281

TH-4097_156102033

TH-4097_156102033