Proceedings of CONF-MLA 2025 Symposium: Applied Artificial Intelligence Research
DOI: 10.54254/2755-2721/2025.BJ26988

Evaluating the Performance Trade-off Between DenseNet and CNN2 for Automatic Modulation Recognition

Xingyu Chen

University of New South Wales, Sydney, Australia
cxy812894778@gmail.com

Abstract. Convolutional neural networks (CNNs) have become the mainstream approach for AMR tasks. However, model selection still faces a trade-off among "performance", "complexity" and "adaptability". Current literature primarily focuses on model innovation itself while lacking systematic comparative analysis of different types of CNN models under the same data conditions. This study investigates the performance differences between two deep neural networks—DenseNet and CNN2—across signal-to-noise ratio (SNR) conditions ranging from 0 dB to 20 dB, based on the standard dataset (RadioML 2018.01A) and evaluates the performance by combining dimensions such as accuracy rate, convergence curve, confusion matrix, etc. The experimental results demonstrate that DenseNet achieves superior recognition accuracy under resource-constrained conditions, particularly at medium-to-high SNR levels. At 20 dB SNR, DenseNet attains an accuracy of 82.1%, outperforming CNN2 by 3.6 percentage points. However, at low SNR, especially at 0 dB, CNN2 demonstrated stronger robustness and generalization ability, but DenseNet was difficult to converge. These findings establish both empirical benchmarks and a theoretical framework for optimizing AMR network architecture selection across diverse communication environments.

Keywords: CNN2, DenseNet, AMR, Convolutional Neural Networks, RadioML 2018.01A

1. Introduction

With the rapid development of wireless communication technology and the sharp increase in the number of devices accessing the network, spectrum resources are becoming increasingly tight. Driven by this trend, communication systems must evolve toward greater intelligence and adaptability. Therefore, automatic modulation recognition (AMR) which is the core technology of cognitive radio, spectrum monitoring and electronic countermeasures is receiving increasing attention. AMR enables receiving terminals to autonomously identify signal modulation schemes without prior information verification. This operation can enhance spectral efficiency. However, traditional AMR methods relying on manual feature extraction and statistical decision-making struggle to adapt to low-SNR and complex channel environments. In recent years, deep learning methods, especially convolutional neural networks (CNN), have emerged. Due to the end-to-end modeling capability of CNN, it has been applied in multiple modulation recognition tasks and made remarkable progress [1,2].

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2025.BJ26988

Currently, automatic modulation recognition methods based on convolutional neural networks (CNNs) have emerged as a focal point of research. A substantial number of scholars have centered their attention on the innovation of network architectures and attempt to improve the model performance and generalization ability through structural optimization. In this study, DenseNet, which is mentioned, enhances the efficiency of information flow through the feature reuse mechanism. However, existing research has predominantly focused on the innovative design of the models themselves and lacks comparative analysis regarding the performance differences of different network architectures under varying signal-to-noise ratio (SNR) scenarios when evaluated under unified data and experimental settings. In real-world communication environments, scenarios with low signal-to-noise ratio (SNR) frequently occur, and available resources are often constrained. Therefore, higher requirements are imposed on the convergence, recognition accuracy, and robustness of models. Currently, there is a lack of research on the training instability of DenseNet under low signal-to-noise ratio (SNR) conditions. In contrast, as a lightweight baseline network, CNN2 demonstrates favorable adaptability and generalization performance under low signal-to-noise ratio [3]. This study is based on the RadioML 2018.01A dataset and selects DenseNet and CNN2 to design training experiments across multiple signal-to-noise ratio (SNR) conditions ranging from 0 dB to 20 dB. Performance comparison and analysis are conducted through confusion matrices, classification accuracy, and training curves. In communication environments where low signal-to-noise ratio (SNR) conditions frequently occur, research on robustness holds significant importance. Consequently, this study provides foundational insights for optimizing and lightweighting network architectures in future AMR systems.

2. Related work

In the existing literature, a large amount of work has focused on the performance and optimization of deep models on standard datasets such as RadioML 2016 and 2018 [4]. Regarding architectural design, each model demonstrates unique characteristics with complementary functional advantages. Traditional architectures like CNN2 serve as prevalent benchmarks due to their structural simplicity, computational efficiency, and training stability [2]. Recent advances incorporate temporal modeling modules to improve feature extraction capability for time-series signal representations. For example, the combination of CNN and GRU can enhance the expression ability of the time structure of modulated signals [5]. There are also studies that optimize network efficiency by introducing a sparse connection mechanism. For example, the SCCNN architecture which can effectively reduce the number of model parameters while maintaining recognition performance proposed by Tunze et al. The proposed SCCNN framework attains 94.39% accuracy on RadioML2018.01A at 20 dB SNR with a 37% reduction in parameter volume compared to baseline models [3].

In addition, multiple studies have attempted to construct more robust models to adapt to multipath fading and low SNR scenarios. For example, Tekbıyık et al. proposed an anti-multipath CNN structure, while Hussein et al. systematically assessed the stability of various deep CNN architectures under complex channel conditions. Collectively, these studies highlight the significant impact of architectural design on model generalization [6,7]. Another development trend is model lightweighting and rapid deployment. Dong et al. introduced a lightweight deep-learning-based AMR method that exhibits strong adaptability in embedded communication devices, owing to its compact model size and consistently high accuracy [8]. As research progresses, some scholars have begun to explore modulation recognition methods that integrate multi-source information and multi-model structures. For instance, Zhang et al. proposed an AMR framework based on multimodal information processing, and Leblebici et al. designed a dedicated CNN architecture for index

DOI: 10.54254/2755-2721/2025.BJ26988

modulation systems [9,10]. Hybrid classifier architectures have recently emerged as a promising research direction. Nasir et al. integrated CNN with a Support Vector Machine (SVM), leveraging their complementary strengths in feature extraction and decision-making to improve classification of low-separability modulation types [11].

In conclusion, current research on automatic modulation recognition (AMR) mainly focuses on the following directions. The first focus is on the innovation of deep neural network structures, such as ResNet, CLDNN and DenseNe [1,3]. The second one is to enhance the robustness of the model under low SNR and complex channels, such as introducing anti-fading mechanisms and multi-path modeling structures [6,7]. The last research direction is to integrate multimodal information and attention mechanisms and improve recognition performance by enhancing the input form or introducing modules such as Transformer [9,12]. Nevertheless, significant gaps remain. While extensive research has concentrated on novel model designs, systematic comparisons—under unified experimental conditions—between lightweight architectures and deeper, reuse-based networks are still lacking.

3. Network architecture design

3.1. CNN2 architecture

CNN2 is a classical shallow convolutional neural network. This network is widely used as a contrast benchmark in the automatic modulation recognition (AMR) task due to the features of simple structure, stable training and high computational efficiency. The network accepts complex signal inputs with dimensions 1×2×1024, where the first dimension denotes batch size (single sample), the second represents I/Q component channels, and the third indicates 1024 temporal sampling points per channel. The first layer of the network is a two-dimensional convolutional layer (Conv2D). This layer employs 256 convolutional kernels (filters) of size 1×3. The function of Conv2D is to extract short-term local correlation features along the time dimension. The feature extraction is concentrated in the time direction, because the convolution kernel does not slide along the frequency domain dimension. This means that the layer can capture the temporal patterns existing in the modulation mode. A subsequent MaxPooling2D layer with 1×2 windows performs temporal downsampling, enhancing feature invariance while reducing computational complexity. The role of pooling is to reduce the number of parameters and computational load. This operation can effectively enhance the invariance and robustness of features. This layer significantly compresses the length of the feature map and enables subsequent layers to focus more on global features during learning. The third layer is the second convolutional layer. The number of channels of this layer is less than that of the first layer. The secondary Conv2D layer hierarchically compresses features while augmenting abstract representations through deeper nonlinear transformations. After this layer, a max pooling of size 1×2 is added to further compress the dimension of the feature map. The multi-channel feature maps are flattened into one-dimensional vectors in the Flatten layer after completing two convolution and pooling operations. The flattened features are processed by a dense layer with ReLU-Softmax cascaded activations, generating probabilistic classification outputs. The overall structure is shown in Figure 1. CNN2 can extract local-to-global discriminative features layer by layer from the input signal and has a strong end-to-end learning ability.

DOI: 10.54254/2755-2721/2025.BJ26988

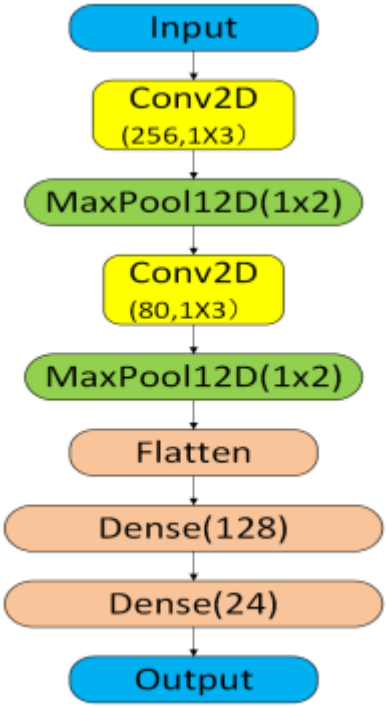

Figure 1. Schematic diagram of CNN2 structure

3.2. DenseNet structure and feature reuse

DenseNet is a deep densely connected convolutional neural network. The core idea of DenseNet is to reuse intermediate-layer features for efficient information flow across depth. The structure of DenseNet is different from the serial connection form between layers in CNN2. DenseNet concatenates all preceding feature maps after every convolution, a mechanism known as feature reuse. The feature reuse mechanism of DenseNet is essentially a cross-level information aggregation strategy. The principle of this mechanism is to jointly model the low-order features extracted by shallow convolution and the high-order abstract features generated by deep convolution. This means that DenseNet is capable of continuously accessing and utilizing the existing feature information during the training process rather than starting from scratch for each layer.

$$x_l = H_l([x_0, x_1, \ldots x_{l-1}]) \tag{1}$$

As shown in Equation 1, the l-th layer receives the concatenated outputs of all preceding layers rather than the single previous layer alone. The main advantage of feature reuse is to enhance the efficiency of information flow. Feature reuse reduces redundant learning and improves the utilization rate and generalization ability of training data.

As shown in Figure 2, The input size of the DenseNet network is the same as that of CNN2 which is a complex signal input of 1×2×1024. The signal is first extracted as initial features through a Conv2D layer with 256 1×3 convolutional kernels. Each subsequent convolutional layer not only processes the output of the previous layer, but also concatenates all the output feature maps of the previous layers and uses them as the input for the next convolutional layer. The network will fuse the features of different scales after completing three consecutive concatenation and convolution operations. Then, the data proceeds to the Dropout layer with a dropout rate of 0.6 to mitigate the

DOI: 10.54254/2755-2721/2025.BJ26988

risk of overfitting. Finally, the data is flattened into a one-dimensional vector through the Flatten layer and then passed into two fully connected layers to complete the classification prediction. The feature reuse mechanism of DenseNet demonstrates excellent adaptability in modulation recognition tasks, especially under medium to high SNR conditions. However, this mechanism also makes the network more prone to accumulating noise redundancy features under low SNR conditions and increases the instability of the training process. Furthermore, as the number of layers in the stitching operation increases, the computational load and memory requirements of the model also rise simultaneously.

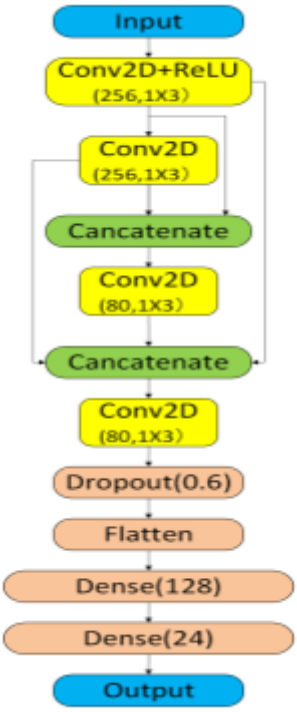

Figure 2. Schematic diagram of DenseNet structure

4. Experimental setup and data processing

Dataset Description: This study employs the RadioML 2018.01A dataset, a benchmark repository for automatic modulation recognition tasks. RadioML 2018.01A includes a total of 24 modulation types, covering various communication standards such as amplitude modulation (AM), frequency modulation (FM), binary frequency shift keying (BFSK), quadrature amplitude modulation (QAM16/QAM64), and phase shift keying (BPSK/QPSK/8PSK/CPFSK). Each complex signal sample comprises 1024 temporal points across I/Q component channels. This dataset includes 26 SNR levels, spanning -20 dB to 30 dB SNR, simulating environments from severe noise degradation to high-fidelity channels [4].

Data Preprocessing: To make the RadioML 2018.01A dataset more suitable for the input format and performance requirements of deep learning model training, this study carried out some preprocessing steps on the original data. The first step is format conversion. The original data is stored in the form of HDF5 files and contains three main fields: X, Y, and Z. The X is the complex signal sample with a dimension of (N, 2, 1024). The Y means one-hot encoded labels with a dimension of (N, 24) and the Z is the SNR value corresponding to each sample. This step involves using the h5py library in Python to load the dataset and convert it into NumPy format for subsequent

DOI: 10.54254/2755-2721/2025.BJ26988

processing. The next steps are the training set division and amplitude normalization. This step involves randomly dividing the dataset into a training set (70%), a validation set (15%), and a test set (15%) for each modulation method and each SNR level. Additionally, amplitude normalization is performed on the I/Q dual-channel signals of each sample to ensure that the signal ranges are uniformly within a similar scale. The final step is dimension reshaping and label encoding. This step converts the original three-dimensional input into a four-dimensional tensor format suitable for the CNN network and keeps the modulation type label Y in one-hot encoding form.

Training Configuration: To ensure that different network structures can be fairly compared under the same conditions, the training parameters were uniformly set in this study. In this study, due to the adaptive learning rate feature that enables rapid convergence in the early stages of training, the Adam optimizer was employed for model training. The initial learning rate is set at 0.001. The two networks were set with the same batch size and training epochs parameters. The batch size is 256 and the number of training epochs is 100. The parameters of batch size and the number of training rounds were set in this way due to the limitations of the experimental platform. The model is implemented using the TensorFlow + Keras framework and runs on a GPU platform that supports CUDA which is the NVIDIA RTX 3070 Ti Laptop. Through the aforementioned training configuration, the model is ensured to be comparable with respect to its architecture, training resource allocation, and evaluation strategies.

5. Experimental results and analysis

5.1. Accuracy comparison across SNR levels

Figure 3 shows the trend of recognition accuracy changes of CNN2 and DenseNet models under different signal-to-noise ratio (SNR) conditions, with the SNR range covering multiple typical communication environments from 0dB to 20dB. Overall, DenseNet achieves superior performance at medium-to-high SNR (≥10 dB), but the robustness of DenseNet under low SNR conditions is slightly insufficient. At the low SNR stage, which is 0 dB and 2 dB, CNN2 is able to maintain relatively stable recognition performance, reaching 58.5% and 58.3% respectively. In contrast, DenseNet failed to converge effectively under the 0dB condition, and its accuracy dropped to 52.0% at 2 dB—underperforming CNN2 by 6.3 percentage points. This means that CNN2 exhibits greater stability and generalization ability in scenarios with severe noise interference, and is suitable for modulation recognition tasks in actual complex environments. As the SNR increases, the accuracy of both approaches has significantly improved. DenseNet achieved 73.1%, 81.3% and 82.1% respectively at 10 dB, 14 dB and 20 dB, all of which were higher than the corresponding values of CNN2 which are 71.4%, 75.2% and 78.5%. The differences were statistically significant ($p < 0.05$), as confirmed by paired t-tests, with 95% confidence intervals showing no overlap in each case. This indicates that DenseNet possesses superior feature extraction capabilities and classification performance under high-quality channel conditions. In conclusion, because of the shallow architecture, CNN2 converges rapidly and has strong training stability under medium and low signal-to-noise ratio conditions. This means that this network is suitable for deployment in communication systems with high requirements for computing resources or prominent real-time performance. Furthermore, due to the deep feature fusion and reuse mechanism of DenseNet, DenseNet demonstrates superior recognition accuracy under high SNR conditions, making it suitable for communication systems with higher precision requirements.

Therefore, resource-constrained edge devices can adopt CNN2 for SNR ≤ 5 dB scenarios, while DenseNet is preferred for high-SNR, high-precision applications.

DOI: 10.54254/2755-2721/2025.BJ26988

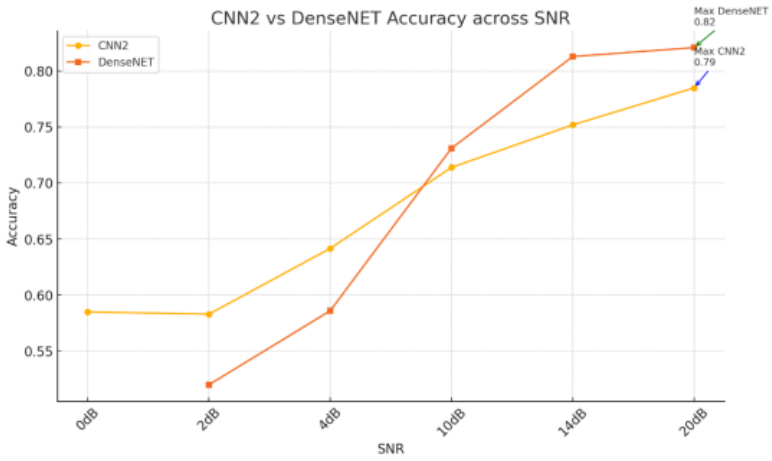

Figure 3. Accuracy comparison between CNN2 and DenseNet

5.2. Confusion matrix and misclassification discussion

As shown in Figures 3 and 4, in order to further compare the classification capabilities of CNN2 and DenseNet in high signal-to-noise ratio scenarios, this study analyzed the confusion matrix when the SNR was 20 dB. Overall, both models can achieve relatively high recognition accuracy for most modulation types, but there are significant differences in their performance for the confusing categories.

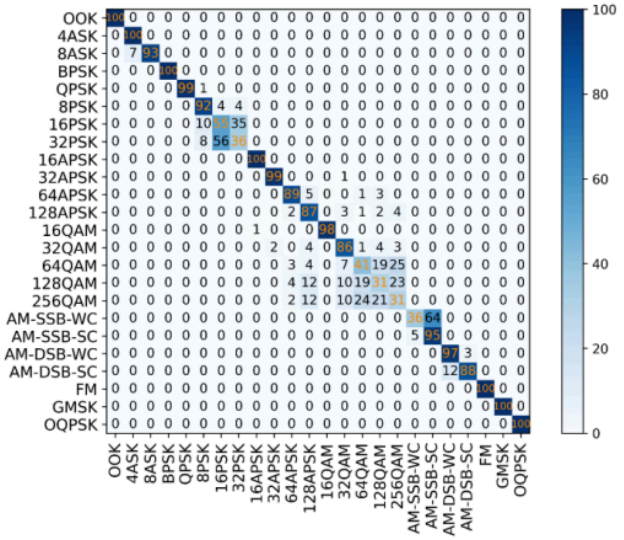

Figure 3. Confusion matrix of CNN2 (SNR=20dB)

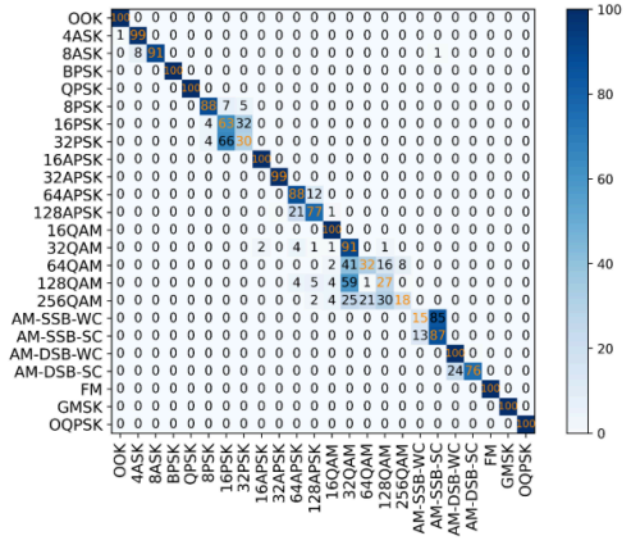

Figure 4. Confusion matrix of DenseNet (SNR=20dB)

From the confusion matrix of DenseNet, it can be observed that most linear modulation schemes such as BPSK, QPSK and OOK can achieve an accuracy rate close to 100%. However, in the higher-order QAM categories such as 64QAM, 128QAM and 256QAM, the model exhibited significant misjudgments. The 64QAM signals were predominantly misclassified as 256QAM (with higher frequency) and 128QAM, indicating a systematic bias toward higher-order modulation misidentification. The 256QAM and128QAM also have the problem of cross-classification errors, such as being wrongly classified as 64QAM or 256QAM. This indicates that even under high signal-

DOI: 10.54254/2755-2721/2025.BJ26988

to-noise ratios, DenseNet still fails to distinguish the boundaries between high-order QAM modulation formats that are similar to each other clearly. This phenomenon may stem from two factors: (a) the inherent similarity in constellation diagrams among high-order QAM schemes, and (b) the feature-sharing mechanism in DenseNet that inadvertently blurs discriminative feature boundaries. It can be observed from Figure 5.2 that the recognition ability of CNN2 is stable, but the accuracy is slightly lower. For the AM-SSB-WC, 256QAM and 128QAM categories, the misjudgment of CNN2 is more obvious. The multi-path modulation types such as AM-SSB-WC have identification blind spots. The 256QAM system achieved only 18% accuracy in classification.

In conclusion, DenseNet's feature multiplexing through dense connectivity may lead to gradient dispersion of higher-order modulation features, instead exacerbating the confusion of similar classes. Although CNN2 exhibits stable training dynamics and robust generalization, its shallow architecture fundamentally restricts discriminative capability for high-order modulations.

5.3. Analysis of DenseNet convergence bottlenecks

Figure 5 shows the confusion matrix of DenseNet under the condition of SNR being 0 dB. The confusion matrix demonstrates that DenseNet exhibits near-complete failure in discriminating modulation schemes at 0 dB SNR. This non-convergence phenomenon in training mainly stems from the failure of the deep-level feature reuse mechanism in the DenseNet structure under low signal-to-noise ratio conditions. At 0 dB, the useful features of the input signal are masked by high-intensity noise, and most of the features extracted by shallow convolution are highly random. Then, DenseNet accumulates and passes these shallow features to deeper layers through skip connections. This leads to the amplification of redundant or invalid information during the feature stacking process. Therefore, the model cannot effectively learn the discriminative features. Furthermore, DenseNet's significantly higher parameter count and denser inter-layer connectivity compared to CNN2 make its gradient updates more sensitive to data distribution. Therefore, the training process of DenseNet is more sensitive to gradient updates and data distribution. In data distributions dominated by high noise, DenseNet is prone to problems such as vanishing gradients or optimization stagnation manifesting as exploding or vanishing gradient norms, as confirmed by gradient L2-norm monitoring. This leads to the network's loss function being unable to decrease, and the accuracy rate remains at a low level for a long time.

DOI: 10.54254/2755-2721/2025.BJ26988

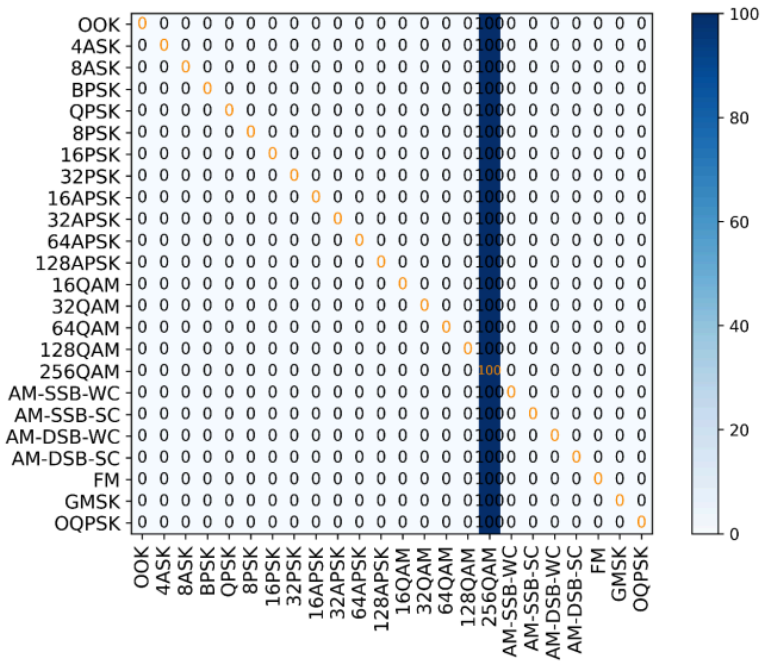

Figure 5. Confusion matrix of DenseNet (SNR=0dB)

6. Conclusion

The experimental results show that DenseNet has a higher accuracy rate under medium and high SNR conditions such as 10 dB, 14 dB, and 20 dB. However, DenseNet fails to converge when training at 0 dB. Although CNN2 has a relatively simple structure, it demonstrates better robustness and training stability under low SNR conditions. Therefore, its accuracy is slightly better than that of DenseNet in high-noise scenarios. In addition, there are some limitations in this study. A key limitation is the insufficient modeling of training-sample noise distributions. The network model lacks an anti-noise mechanism which limits its practicality in complex channels. In addition, the balance of the dataset and the differences between the experimental settings and the real scenarios may also affect the generalization ability of the model performance. Subsequent research can optimize the network structure from multiple directions. The first is that noise-resistant modules can be introduced such as attention mechanisms, residual channel selection, and noise-robust enhancement loss functions. Another way is model distillation and feature re-weighting can be utilized to enhance its adaptability under low SNR conditions. In addition, it is possible to further explore the integration of more network structures such as ResNet, CLDNN, or hybrid classification strategies to enhance generalization ability and stability in different channel environments.

References

[1] F. Zhang, C. Luo, J. Xu, Y. Luo and F.-C. Zheng, "Deep Learning Based Automatic Modulation Recognition: Models, Datasets, and Challenges, " Digital Signal Processing, vol. 129, p. 103650, 2022.

[2] Xiao, W., Luo, Z., & Hu, Q. (2022). 'A Review of Research on Signal Modulation Recognition Based on Deep Learning'. Electronics, 11(17), 2764. https: //doi.org/10.3390/electronics11172764

[3] G. B. Tunze, T. Huynh-The, J.-M. Lee, and D.-S. Kim, 'Sparsely Connected CNN for Efficient Automatic Modulation Recognition, ' IEEE Trans. Veh. Technol., vol. 69, no. 12, pp. 15557–15570, Dec. 2020.

[4] RadioML Datasets. [Online]. Available: https: //www.kaggle.com/datasets/pinxau1000/radioml2018?resource=download

DOI: 10.54254/2755-2721/2025.BJ26988

[5] F. Liu, Z. Zhang and R. Zhou, 'Automatic modulation recognition based on CNN and GRU, ' Tsinghua Science and Technology, vol. 27, no. 2, pp. 422-431, April 2022, doi: 10.26599/TST.2020.9010057.
[6] K. Tekbıyık, A. R. Ekti, A. Görçin, G. K. Kurt and C. Keçeci, 'Robust and Fast Automatic Modulation Classification with CNN under Multipath Fading Channels, ' 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), Antwerp, Belgium, 2020, pp. 1-6, doi: 10.1109/VTC2020-Spring48590.2020.9128408.
[7] H. S. Hussein, M. H. Essai Ali, M. Ismeil, M. N. Shaaban, M. L. Mohamed and H. A. Atallah, 'Automatic Modulation Classification: Convolutional Deep Learning Neural Networks Approaches', in IEEE Access, vol. 11, pp. 98695-98705, 2023, doi: 10.1109/ACCESS.2023.3313393.
[8] Dong YI, Di WU, Tao HU, 'A Lightweight Automatic Modulation Recognition Algorithm Based on Deep Learning', IEICE TRANSACTIONS on Communications, vol. E106-B, no. 4, pp. 367-373, April 2023, doi: 10.1587/transcom.2022EBP3087.
[9] Zhang, W., Xue, K., Yao, A., & Sun, Y. (2024). 'Automatic Modulation Recognition Based on Multimodal Information Processing: A New Approach and Application'. Electronics, 13(22), 4568. doi: https: //doi.org/10.3390/electronics13224568
[10] M. Leblebici, A. Çalhan and M. Cicioğlu, 'CNN-based automatic modulation recognition for index modulation systems, ' Expert Systems with Applications, vol. 240, 2024, Art. no. 122665. doi: 10.1016/j.eswa.2023.122665.
[11] S. Nasir, S. A. Sheikh and F. M. Malik, 'Automatic modulation classification using convolutional neural network and support vector machine, ' Digital Signal Processing, vol. 164, 2025, Art. no. 105249. doi: 10.1016/j.dsp.2025.105249.
[12] S. Ansari et al., 'Attention-Enhanced Hybrid Automatic Modulation Classification for Advanced Wireless Communication Systems: A Deep Learning-Transformer Framework, ' in IEEE Access, vol. 13, pp. 105463-105491, 2025, doi: 10.1109/ACCESS.2025.3580574.