Automatic Modulation Recognition Algorithm Based on Multiscale Feature Extraction and Improved Encoder Structure

Tianfei Zhang[1], Yecai Guo[1,2], Jiao Ding[1], Haiyan Long[1], Lei Zhang[1,*]
[1]School of Electrical and Electronic Engineering, Anhui Institute of Information Technology, No. 1 Yonghe Rd,241000 Wuhu, China
[2]School of Electronics and Information Engineering, Nanjing University of Information Science & Technology, No. 219 Ningliu Rd, 210044 Nanjing, China
2016007@aiit.edu.cn; guoyecai@nuist.edu.cn; 2015014@aiit.edu.cn*; 2013050@aiit.edu.cn; *2013033@aiit.edu.cn*

***Abstract*—Automatic modulation recognition (AMR) technology is crucial in modern communication systems. To enhance the recognition performance of wireless communication systems for modulation signals, a method based on multiscale feature extraction and an improved encoder structure is proposed. First, in this proposed method, in the data preprocessing stage, phase transformation (PT) is used to eliminate the influence of phase offset. Second, the multiscale feature extraction (MSFE) module is utilised to capture both local details and global structural features of the signal, thereby reducing the information loss rate. Third, the autoencoder (AE) module preserves temporal information, reconstructs features, and suppresses noise, significantly improving the feature reconstruction and recognition capabilities of modulation signals while minimising reconstruction loss. Simulation results have demonstrated that the proposed algorithm can achieve an average recognition accuracy of 92.5 % in the RML2016.10A, which has an improvement of about 2 % to 11 % in comparison with other mainstream algorithms in average recognition accuracy, can obtain peak recognition accuracies of 93.5 % and 93.9 % in the RML2016.10A and RML2016.10B datasets, respectively, within a signal-to-noise ratio (SNR) range of 0 dB to 18 dB, thus demonstrating its excellent recognition accuracy and robustness.**

***Index Terms*—Automatic modulation recognition; Phase transformation; Multiscale feature extraction module; Convolutional neural network; Encoders; Autoencoder.**

I. INTRODUCTION

In the real world, communication environments are subject to various interference and noise conditions. Automatic modulation recognition (AMR) technology [1], [2] is widely used to identify the modulation scheme of communication signals, as well as in signal monitoring, fault detection, and security assurance. Therefore, AMR technology has attracted widespread attention to improve the stability and reliability of communication systems.

In conventional AMR methodologies, the likelihood-based (LB) approach [3] typically requires a known modulation scheme to be set in advance, and the demodulator only performs signal demodulation based on the predefined modulation scheme. Feature-based (FB) methods [4] rely on manual design and extraction of signal features, and their performance is heavily dependent on the selected set of features and often degrades under varying channel conditions. In contrast, deep learning techniques can offer greater simplicity, flexibility, and generalisation capability, and are capable of handling complex and dynamic communication environments. With the development of deep learning technology, especially the successful application of convolution neural network (CNN) [5] and recurrent neural network (RNN) [6], the AMR field has undergone tremendous changes. In 1996, Azzouz and Nandi [7] mainly studied the AMR method for communication signals, Artificial Neural Networks (ANN) were first applied to the automatic modulation recognition of communication signals. In 2016, O'Shea, Corgan, and Clancy [8] pioneered the application of CNNs to AMR, which achieved better results than traditional artificial feature extraction methods, but this network architecture exhibited limited discriminative capacity for higher-order modulation signals. Zhang, Luo, Wang, Gan, and Xiang [9] proposed a hybrid CNN-LSTM model that comprehensively considers both interactive features among signals and their spatio-temporal characteristics, but does not take into account differential features in multichannel signal components. Yang, Yang, Feng, Hao, and Wang [10] incorporated an attention mechanism (AM) with CNN to enable the model to automatically learn with the highest weight, thus effectively improving model performance. Nevertheless, this network architecture relies on a strictly stacked multilayer convolutional structure, which limits feature extraction due to inherent sequential dependencies. Shi, Xu, Jiang, and Qi [11] adopted a parallel combination of CNN and LSTM encoders to extract spatio-temporal features from normalised amplitude and phase (AP) data to achieve high recognition accuracy at low signal-to-noise ratios with relatively low computational cost. However, this method does not consider the impact of phase offset.

Manuscript received 23 August, 2025; accepted 24 October, 2025.
This research was supported by the Key Natural Science Research Project in Universities of Anhui Province under Grants Nos. 2023AH052917 and . 2024AH050639; by the Excellent Young Teachers Cultivation Program of Anhui Province under Grant No. YQYB2025079.

On the basis of the above analyses, this paper proposes a hybrid network approach based on a multiscale feature extraction module (MSFE) and an improved encoder structure. First, the proposed method employs a phase transformation (PT) module to eliminate the phase offset effect on the input P-component. Second, the MSFE module is used to extract and preserve multiscale differential features and interactive features. Third, an autoencoder (AE) module is adopted to compress dimensions while retaining key features, and to reconstruct redundancy-free signals as supplementary feature details, thereby effectively enhancing the recognition performance of the proposed algorithm. Experiments have demonstrated that the proposed algorithm can achieve excellent recognition performance under low parameter scale conditions.

II. SIGNAL MODEL

In communication systems, the frequency spectrum of baseband signals such as speech and images are typically concentrated in the low-frequency range near zero frequency, making the signals more susceptible to negative effects such as noise, nonlinear distortion, and multipath interference during transmission, leading to severe distortion of the receiving signal. Therefore, to enhance the anti-interference capability and transmission efficiency of the signal during transmission, the transmitting end usually needs to modulate the signal, i.e., load the baseband signal onto a high-frequency carrier. The formula of the specific modulation is as follows

$$x(t) = \alpha(t)e^{j(2\pi f_0 + \varphi_0)} \times s(t) + n(t), \tag{1}$$

where $x(t)$ is the received signal, $s(t)$ denotes the transmitted signal, $\alpha(t)$ is recorded as the complex channel gain, f_0 means the frequency offset, φ_0 represents the phase offset, and $n(t)$ stands for zero-mean additive white Gaussian noise.

In-phase and quadrature (IQ) data can be converted into amplitude and phase (AP) data using the following equations:

$$A = \sqrt{I^2 + Q^2}, \tag{2}$$

$$P = \arctan(Q / I), \tag{3}$$

where A and P represent the amplitude component and phase component of the AP sample signal, respectively.

In fact, the essence of modulation lies in encoding information into the amplitude and phase of signal waveforms. This transformation can more intuitively reveal the intrinsic differences between different modulation types. By amplifying the differences among various types of modulation, it helps the receiver to accurately distinguish and recover different signals in mixed channels [11].

III. FUNDAMENTALS

1. *Experimental Datasets*

The experiments were carried out on two public datasets, i.e., RML2016.10A and RML2016.10B. The RML2016.10A dataset comprises 11 modulation schemes, including BPSK, QPSK, 8PSK, 16-QAM, 64-QAM, CPFSK, GFSK, 4-PAM, AM-DSB, AM-SSB, and WBFM, with a total of 220,000 signal samples. In contrast, the RML2016.10B dataset contains 10 other types of modulation, except for the AM-SSB analogue modulation mode, with 1.2 million samples. The signal-to-noise ratio range of these two datasets is between -20 dB to 18 dB, comprehensively considering real-world channel impairments, such as additive white Gaussian noise (AWGN), selective fading, carrier frequency offset, multipath propagation, and sampling rate offset.

2. *Data Preprocessing*

On the RML2016.10A dataset, the recognition performance of two classical models, CNN2 [12] and LSTM [13], was evaluated using raw IQ samples and amplitude-phase (AP) samples as input signals, respectively. The results are illustrated in Fig. 1. When the LSTM model processed raw IQ-format data, its recognition accuracy dropped sharply from 90 % to 60 % within the SNR range in [0, 18] dB. This indicates that the complexity of IQ signals, represented as complex-valued two-dimensional time-series data, makes it difficult for lightweight LSTM models to effectively capture temporal characteristics and correlations in the signal. After converting the input data to AP format, CNN2 has exhibited a slight performance degradation under low SNR conditions, while LSTM has demonstrated significant improvement throughout the entire SNR range. These results indicate that AP-format data make the prominent features of the signal more compact and easier to process, while also being more conducive to the model learning the long-term temporal dependencies of the signal in lightweight situations. Based on these findings, this study adopts AP-format data that are used as the input signal.

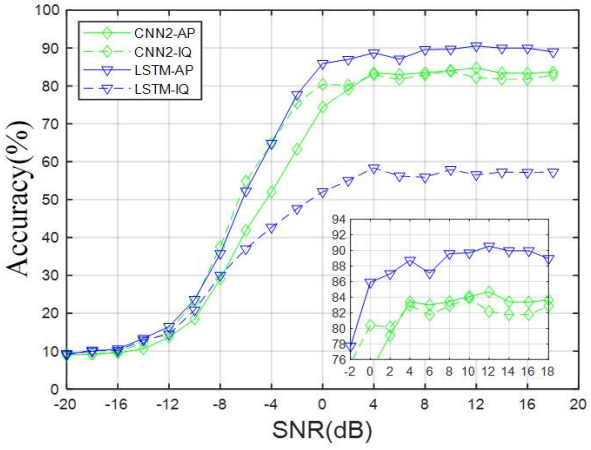


Fig. 1. Recognition accuracy before and after data preprocessing.

Furthermore, during wireless transmission, signals are prone to phase offsets, which can alter the positions of signal constellation points [14], [15], thus leading to misjudgment of signal modulation modes at the receiving end. In the IQ coordinate system, the phase offset can be expressed as:

$$\begin{bmatrix} I \\ Q \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} I' \\ Q' \end{bmatrix}, \tag{4}$$

where I and Q represent the coordinates after phase offset, I' and Q' denote the original coordinates before phase

offset, and $\begin{bmatrix}\cos\theta & -\sin\theta \\ \sin\theta & \cos\theta\end{bmatrix}$ represents the offset matrix. Here, we denote the offset matrix as $\boldsymbol{C}$.

Combining with (3), the elimination of phase offset effects in AP-format data can be expressed as:

$$A' = \sqrt{I^2 + Q^2}, \tag{5}$$

$$P' = \arctan\left(\mathbf{C}[0]\begin{bmatrix}Q \\ I\end{bmatrix} \Big/ \mathbf{C}[1]\begin{bmatrix}Q \\ I\end{bmatrix}\right), \tag{6}$$

where A' and P' represent the AP-format data without phase offset, $\boldsymbol{C}[0]$ denotes the parameters of the first row of the offset matrix, $\boldsymbol{C}[1]$ indicates the parameters of the second row of the offset matrix, and $\boldsymbol{C}[0]/\boldsymbol{C}[1]$ is referred to as the anti-offset factor.

It can be seen from (5) and (6) that the component A is not affected by phase offset, so it can be input directly into the network after format conversion. However, the component P is affected by phase offset and can be input into the network after the influence is eliminated through the phase compensation function of the PT module.

The structure of the PT module is shown in Fig. 2. This module first utilises a flatten layer and a fully connected layer to obtain the phase offset parameter φ_0 through extensive sample training. Specifically, the specific value of the anti-offset factor $\boldsymbol{C}'[0]/\boldsymbol{C}'[1]$ can be derived via the offset matrix $\boldsymbol{C}' = \begin{bmatrix}\cos\varphi_0 & -\sin\varphi_0 \\ \sin\varphi_0 & \cos\varphi_0\end{bmatrix}$, furthermore, input components without the influence of phase offset can be obtained.

3. *Implementation Details*

The model was trained using classification cross-entropy as the loss function, and the batch size is set to 400 to avoid getting stuck in local optima of the algorithm. The initial learning rate was set to 1e-3 with a minimum threshold of 1e-7. When the validation loss does not decrease for eight consecutive epochs, the learning rate will be halved to accelerate the convergence of the algorithm to the optimal solution. In addition, to mitigate overfitting, an early stopping mechanism was implemented. When the validation loss does not decrease further within 20 consecutive epochs, the training will terminate prematurely. All experiments were carried out on a GeForce GTX 1080Ti GPU using the Keras deep learning library within the TensorFlow framework.

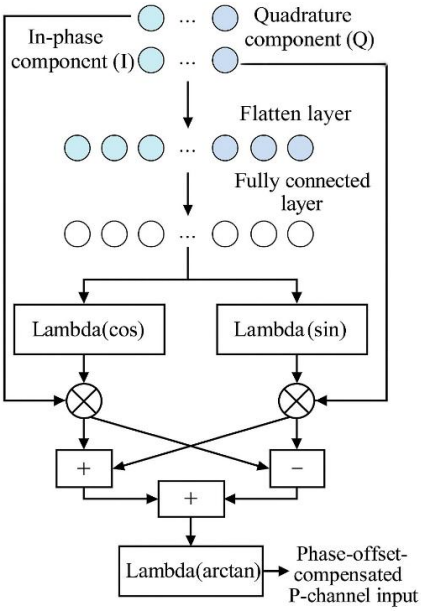


Fig. 2. PT module.

IV. MODEL DESIGN

To effectively capture both local and global features of signals while improving recognition accuracy and robustness in complex channel environments, the proposed algorithm incorporates the multiscale feature extraction (MSFE) module and the autoencoder (AE) module. The detailed architecture of the constructed algorithm is illustrated in Fig. 3, and the network parameters are summarised in Table I.

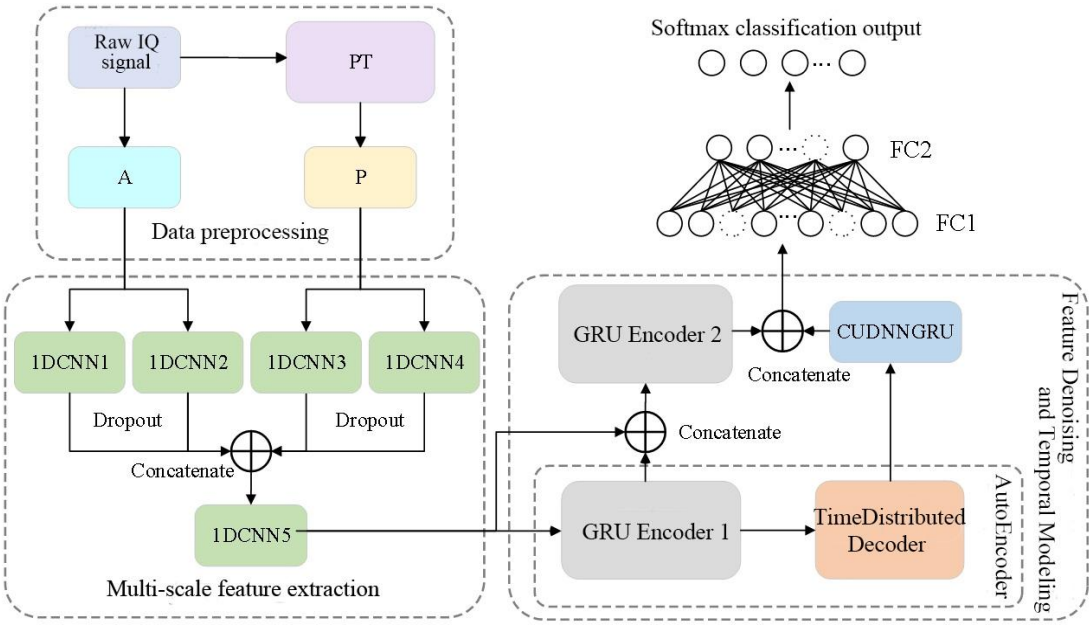


Fig. 3. Structure of the MSFE-AE model.

TABLE I. CONFIGURATION OF THE PROPOSED NETWORK ARCHITECTURE.

Layer	Output dimension	Configuration parameters
A	128×1	——
P	128×1	——
1DCNN1	128×16	Number of convolution kernels: 16 Convolution kernel size: 5 Activation function: ReLU
1DCNN2	128×16	Number of convolution kernels: 16 Convolution kernel size: 8 Activation function: ReLU
1DCNN3	128×16	Number of convolution kernels: 16 Convolution kernel size: 5 Activation function: ReLU
1DCNN4	128×16	Number of convolution kernels: 16 Convolution kernel size: 8 Activation function: ReLU
Concatenate	128×64	(1DCNN1: 1DCNN2: 1DCNN3: 1DCNN4)
1DCNN5	128×16	Number of convolution kernels: 16 Convolution kernel size: 8 Activation function: ReLU
GRU Encoder1 (CUDNNGRU1 CUDNNGRU2)	128×32 128×16	32 units 16 units
Concatenate	128×32	(GRU Encoder1(CUDNNGRU2): 1DCNN5)
GRU Encoder 2 (CUDNNGRU1 CUDNNGRU2)	128×64 32	64 units 32 units
TimeDistibuted (Dense) Encoder	128×32	32 units
CUDNNGRU	32	32 units
Concatenate	64	(GRU Encoder2(CUDNNGRU2): CUDNNGRU)
FC1	32	32 units
FC2	16	16 units
Output (Softmax)	n	n-Type Modulated Signals

1. MSFE module

For the input sequence $\boldsymbol{x}=[x_1,x_2,\ldots,x_n]$, when the convolution kernel is denoted as $\boldsymbol{W}=[W_1,W_2,\ldots,W_k]$, the one-dimensional convolution operation can be expressed as

$$y_i=\sum_{j=1}^{k}\boldsymbol{W}_j\boldsymbol{x}_{i+j-1},i=1,2,\ldots,n-k+1, \tag{7}$$

where $\boldsymbol{W}$ represents the convolution kernel, k denotes the convolution kernel size, n and y_i are the sequence length and the output, respectively.

The multiscale one-dimensional convolution can be formulated as

$$\boldsymbol{y}_{\text{concat}}=[y_1,y_2,\ldots,y_i],i=1,2,\ldots,n, \tag{8}$$

where $\boldsymbol{y}_{\text{concat}}$ represents the output of the multiscale one-dimensional convolution module and n denotes the number of one-dimensional convolutions.

The multiscale feature extraction (MSFE) module takes *A* and *P* dual-channel data as input, which can avoid ignoring the differential features of amplitude or phase when using AP-format data as a single-channel input. It employs four multiscale one-dimensional convolution modules for processing the input data in parallel, and mainly addressing the limitation of single-scale convolution kernels in feature extraction. This enables the proposed algorithm to simultaneously capture both local details and global structural features of the signal to avoid feature loss and improve feature diversity. Multiscale parallel processing can also improve the robustness of the proposed algorithm against noise, distortion, and channel interference, and reduce information loss, as well as maintain high parameter efficiency through weight sharing and parallel computing. This design allows the proposed algorithm to fully cover the multiscale features of the signal, thereby improving the performance and robustness in complex signal modulation recognition tasks.

2. AE module

The feature denoising and temporal modelling module consists mainly of an autoencoder (AE) [16], [17], and an encoder [18], [19]. Its detailed architecture is illustrated in Figs. 4 and 5. Among them, the autoencoders are responsible for feature learning and reconstruction, while the encoders perform multiscale feature extraction and dimensional compression.

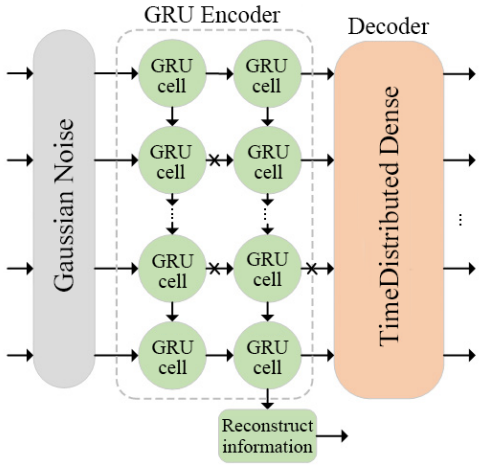


Fig. 4. Autoencoder structure.

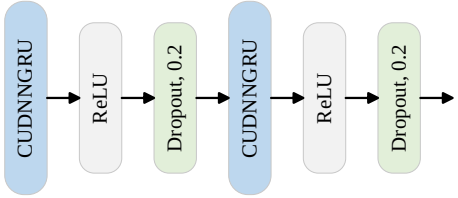


Fig. 5. Structure of the GRU encoder.

The GRU encoder comprises two stacked CUDNNGRU layers and two Dropout layers. The CUDNNGRU layers process sequential data through a recurrent neural network structure, whose core idea is to control the flow of information through gating mechanisms. The computational logic can be formally expressed as:

$$z_t = \sigma(W_z \times [h_{t-1}, x_t]), \tag{9}$$

$$r_t = \sigma(W_r \times [h_{t-1}, x_t]), \tag{10}$$

$$\tilde{h}_t = \tanh(W_{\tilde{h}} \times [r_t \times h_{t-1}, x_t]), \tag{11}$$

$$h_t = z_t \times \tilde{h}_t + (1 - z_t) h_{t-1}, \tag{12}$$

where x_t is the current unit input, h_t represents the current unit output, h_{t-1} denotes the unit output at the previous time step, z_t stands for the update gate, r_t is regarded as the reset gate, W_0 means the weight matrix, and $\sigma(\cdot)$, $\tanh(\cdot)$ represent the sigmoid and hyperbolic tangent functions, respectively.

The Dropout layer [20], [21] effectively prevents overfitting during training by randomly deactivating a portion of the GRU units, thus mitigating excessive reliance on any individual unit. Its mask generation process can be formally expressed as

$$m_t = \begin{cases} 1 & \text{retain with probability p,} \\ 0 & \text{retain with probability 1-p.} \end{cases} \tag{13}$$

The output after Dropout processing is given by

$$h_t^{'} = m_t \odot h_t / p, \tag{14}$$

where p is the retention probability and $h_t^{'}$ is the hidden state after Dropout.

The GRU encoder projects high-dimensional multiscale features into low-dimensional feature sequences [22], [23] to effectively eliminate redundancy and noise interference in the high-dimensional features while facilitating the identification and extraction of crucial high-weight features from modulated signals.

The decoder employs a TimeDistributed Dense architecture, in which two encapsulated fully connected layers with 32 neurons each are included. The TimeDistributed layer ensures temporal correlation between time steps and avoids the loss of time dimension information by applying same weight matrix and bias vector to all time steps. In the process of feature reconstruction, t nonlinear transformations of feature space is achieved through a fully connected layers, and its mathematical expression is written as

$$y_t = W_d h_t + b_d, \tag{15}$$

where W_d and b_d are the weight matrix and bias vector of the decoder, respectively, y_t is the output at the t^{th} time step.

The reconstruction loss function of the entire AE module is typically defined as the mean squared error (MSE) and expressed as

$$L = \frac{1}{T} \sum_{t=1}^{T} (x_t - y_t')^2, \tag{16}$$

where T represents the length of the sequence, x_t denotes the input at the t^{th} time step, and y_t' corresponds to the reconstructed output at the t^{th} time step. By minimising the reconstruction loss function L, the AE module can effectively extract high-weight features from the input sequence and leach redundant information and noise during the reconstruction process.

The AE module can learn and efficiently reconstruct signal features through an autoencoder, while utilising a GRU encoder to map high-dimensional multiscale features into low-dimensional sequences, eliminating redundant information and noise interference, and extracting key high-weight features from modulated signals. The decoder adopts a TimeDistributed Dense structure to ensure the correlation between time steps, avoid the loss of temporal dimension information, and realises a nonlinear transformation of the feature space through fully connected layers. Ultimately, while minimising reconstruction loss, the module achieved efficient feature extraction, noise suppression, and preservation of temporal dimension information, significantly enhancing the feature reconstruction and recognition capabilities of modulated signals.

V. EXPERIMENTAL RESULTS AND ANALYSES

1. *Performance Analyses of Data Preprocessing*

Based on the network parameter configuration established in Section III, experiments were conducted to verify the recognition performance of the classical LSTM model before and after our data preprocessing, using AP-format data as input. For clarity in distinction, the preprocessed model is named PC-LSTM.

The comparison of recognition accuracy between the PC-LSTM and the LSTM model is shown in Fig. 6, and the simulation results of both are presented in Table II.

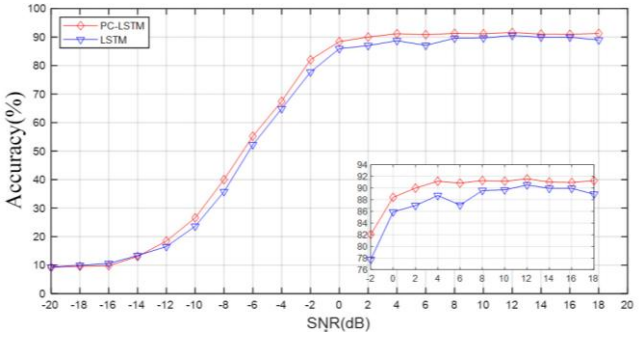


Fig. 6. The recognition accuracy of PC-LSTM and LSTM.

TABLE II. SIMULATION RESULTS OF PC-LSTM AND LSTM.

Model	Peak recognition accuracy	Average recognition accuracy in high-SNR regime	Overall mean recognition accuracy
PC-LSTM	**91.58 %**	**90.82 %**	**61.97 %**
LSTM	90.54 %	87.94 %	60.07 %

Experimental results indicate that after the P input component in the AP format data undergoes phase correction using the phase compensation factor in the data preprocessing stage, the recognition accuracy of the LSTM model is significantly improved. Specifically, the highest recognition accuracy of the PC-LSTM model reaches 91.58 %, which is 1 % higher than 90.54 % of the LSTM model. Within the high SNR range (SNR ≥ 0 dB), the average recognition accuracy of the PC-LSTM model is 90.82 %, which is approximately 3 % higher than 87.94 % of the LSTM model. In terms of overall average recognition accuracy, the PC-LSTM model achieves 61.97 %, which is about 2 % higher than 60.07 % of LSTM. These results have verified the effectiveness of data preprocessing.

2. Comparative Experiments

To demonstrate the performance advantages of our proposed MSFE-AE model, we conducted extensive comparative experiments on mainstream models, such as MCLDNN [24], LSTM, CNN2, CNN-LSTM [9], and BMCCLDNN [25], using both RML2016.10A and RML2016.10B datasets, as shown in Fig. 7(a). The experimental results reveal that our model achieves an average recognition accuracy of 92.5 % within the range of the signal-to-noise ratio (SNR) of 0 dB–18 dB, which represents an improvement of about 2 % to 11 % compared to other mainstream models. Notably, at SNR = 18 dB, our model attains peak performance levels of 93.5 % on RML2016.10A and 93.9 % on RML2016.10B, demonstrating excellent recognition accuracy and operational robustness.

The computational efficiency metrics presented in Table III show that our model has the lowest number of parameters compared to other models, significantly reducing memory requirements and optimising space complexity. Although the training time of the lightweight LSTM and BMCCLDNN models is slightly shorter, their recognition accuracy is 3 % and 4 % lower than our model, respectively. However, MCLDNN, CNN-LSTM, and CNN2 with larger parameter scales not only have higher training time, but also lag behind in recognition accuracy. These comprehensive evaluations confirm that the MSFE-AE model performs outstandingly in balancing recognition accuracy and computational efficiency, combining high-performance capabilities with operational practicality.

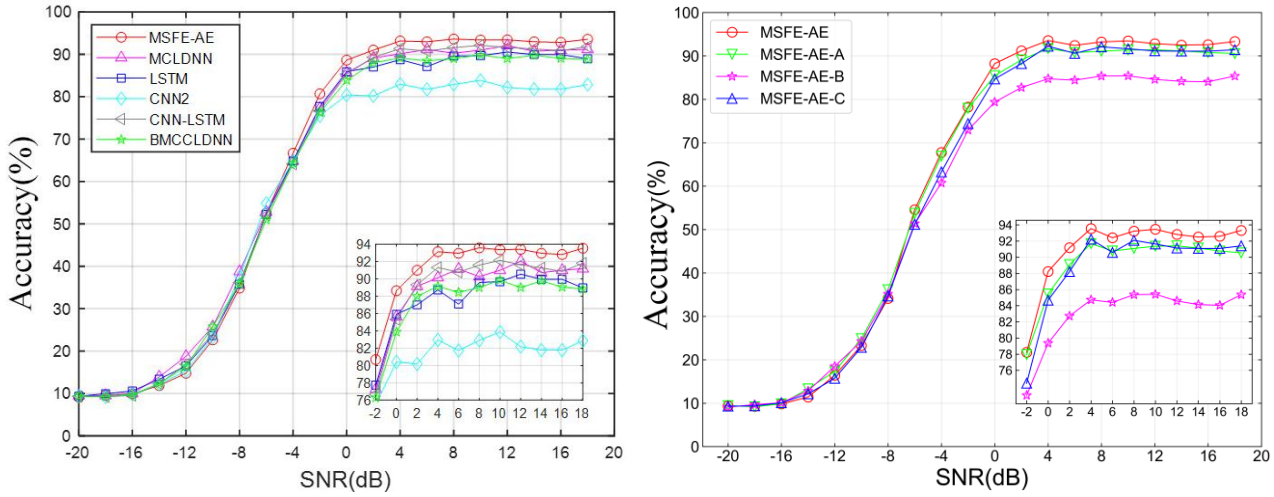


Fig. 7. Recognition accuracy of (a) and (b) in the RML2016.10A dataset: (a) Comparative experiments; (b) Ablation studies.

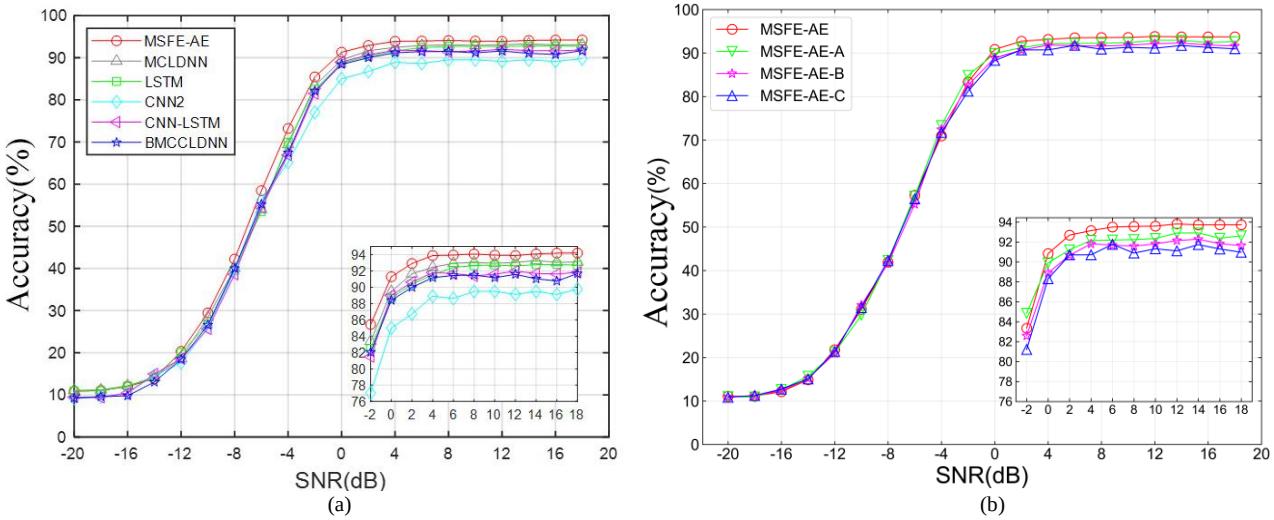


Fig. 8. Recognition accuracy of (a) and (b) in the RML2016.10B dataset: (a) Comparative experiments; (b) Ablation studies.

3. Ablation Studies

To verify the effectiveness of the core modules, ablation experiments were designed, as shown in Fig. 7(b) and Fig. 8(b). The following variants of the model were constructed for ablation experiments: MSFE-AE-A (removing PT module d), MSFE-AE-B (removing MSFE

module), and MSFE-AE-C (removing AE module r).

The statistical results of the time costs required to train the variants in Table III indicate that after removing any module, the number of parameters and the time of a single iteration of the variant have decreased, but the number of iterations has increased significantly. Based on Fig. 7(b), the average recognition accuracy of the variants across the entire SNR range is substantially reduced. With the SNR is in the range of 0 dB~18 dB, the average recognition accuracy of MSFE-AE-A, MSFE-AE-B, and MSFE-AE-C decreases by 2.5 % to 9 % respectively.

TABLE III. COMPARISON OF TRAINING TIME COSTS OF DIFFERENT MODELS ON THE RML2016.10A DATASET.

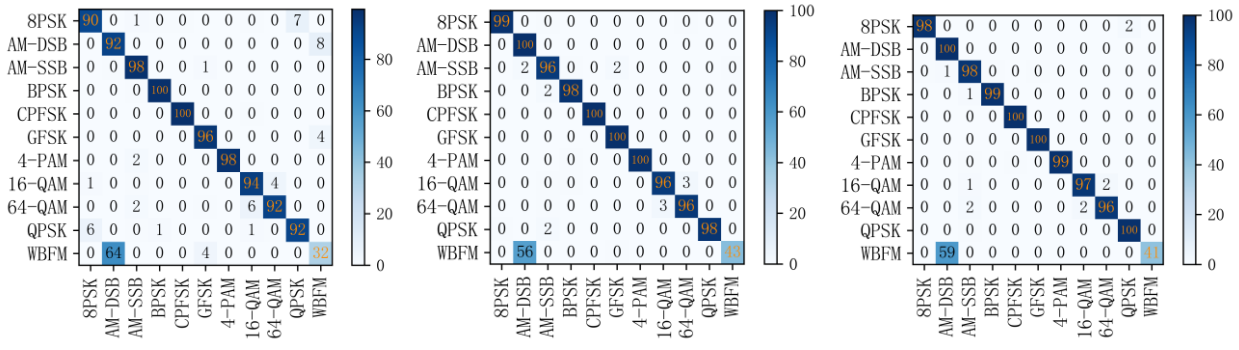
Metrics / Model	Parameters	Iter (s)	Time/Iter (s)	Total Time (s)
MCLDNN	406 070	128	18	2 304
CNN-LSTM	1 273 443	165	58	9 570
CNN2	858 123	201	31	6 231
LSTM	200 970	96	11	1 056
BMCCLDNN	160 569	124	14	1 736
MSFE-AE	**58 652**	**104**	**17**	**1 768**
MSFE-AE-A	58 395	128	16	1 872
MSFE-AE-B	38 252	176	7	1 232
MSFE-AE-C	41 692	135	15	2 025

This demonstrates that the PT module improves the model's robustness to signal phase changes by eliminating the influence of phase offsets; The MSFE module enhances the model's ability to express multiscale features by capturing local details and global structural features of signals; The AE module further optimises the discriminative and anti-interference capabilities of features through time information preservation, feature reconstruction, and noise suppression. The synergistic effect of the three enables the model to achieve high recognition accuracy and stability in complex channel environments.

4. Confusion Matrix Analysis

To gain a clearer and more intuitive understanding of the performance of our model and various misclassification situations, a confusion matrix is used to further evaluate the performance of our model. A confusion matrix is a tool for assessing the performance of recognition models, where columns correspond to predicted labels and rows correspond to actual labels. The values in the matrix represent prediction probabilities, while the diagonal elements indicate accuracy. The confusion matrices of the model in the RML2016.10A dataset under signal-to-noise ratios (SNRs) of 0 dB, 10 dB, and 18 dB are shown in Fig. 9. Analyses of Fig. 9 reveal that under high SNR conditions, the model can achieve a recognition accuracy of more than 92 % for most signals, with significant confusion occurring only in the recognition of the WBFM modulated signals.

In addition, the model exhibits an excellent discriminative ability for 16-QAM and 64-QAM modulation, which is difficult for most models to distinguish. The reason why WBFM and AM-DSB modulations are difficult to distinguish lies in the fact that both are generated by sampling analogue audio signals and the amplitude maps of the two signals are highly similar, especially during audio silent periods [26], which further reduces the difference in features between the two types of signals and thus increases the difficulty of accurate recognition.



(a) (b) (c)

Fig. 9. Confusion matrix of the MSFE-AE model under: (a) SNR = 0 dB; (b) SNR = 10 dB; (c) SNR = 18 dB.

VI. CONCLUSIONS

To enhance the recognition performance of modulation signals in wireless communication systems, a hybrid neural network model is proposed. The proposed model first eliminates the influence of phase offset on the phase input component (P) through a phase transformation module. Next, the multiscale feature extraction module is employed for capturing amplitude and phase features at different scales, thereby reducing information loss. Simultaneously, the autoencoder and encoder modules compress the high-dimensional features of modulation signals into low-dimensional representations, preserving key feature information, and filtering out redundant information and noise through signal reconstruction to improve signal clarity and recognition accuracy. Subsequently, by minimising the reconstruction loss function, the proposed model can effectively identify and extract key high-weight features from the modulated signal, thus significantly boosting the recognition performance.

Experimental results demonstrate that the proposed model achieves an average recognition accuracy of 92.5 % on the publicly available dataset RML2016.10A within the range of the signal-to-noise ratio of 0 dB to 18 dB, which is 2 % to 11 % higher than other mainstream models. In addition, the generalisation experiment on the RML2016.10B dataset also yields excellent performance, enabling accurate recognition of all modulation types, except WBFM.

Although the proposed model has achieved significant improvements in modulation recognition accuracy and generalisation, breakthroughs are still needed in aspects such as recognising phase-sensitive modulation signals and

adapting to dynamic environments. In the future, novel signal processing algorithms (e.g., domain adaptation) and deep learning optimization strategies (e.g., attention mechanisms) can be integrated to further enhance the robustness and practicality of the model.

ACKNOWLEDGMENT

We thank the editor and the anonymous reviewers for their valuable suggestions on improving this paper.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] Y. Qu, Z. Lu, R. Zeng, J. Wang, and J. Wang "Enhancing automatic modulation recognition through robust global feature extraction", *IEEE Transactions on Vehicular Technology*, vol. 74, no. 3, pp. 4192–4207, 2025. DOI: 10.1109/TVT.2024.3486079.

[2] N. Rashvand *et al.*, "Enhancing automatic modulation recognition for IoT applications using transformers", *IoT*, vol. 5, no. 2, pp. 212–226, 2024. DOI: 10.3390/iot5020011.

[3] P. Ghasemzadeh, S. Banerjee, M. Hempel, and H. Sharif, "Performance evaluation of feature-based automatic modulation classification", in *Proc. of 2018 12th International Conference on Signal Processing and Communication Systems (ICSPCS)*, 2018, pp. 1–5. DOI: 10.1109/ICSPCS.2018.8631742.

[4] S. Huang, Y. Yao, Z. Wei, Z. Feng, and P. Zhang, "Automatic modulation classification of overlapped sources using multiple cumulants", *IEEE Transactions on Vehicular Technology*, vol. 66, no. 7, pp. 6089–6101, 2017. DOI: 10.1109/TVT.2016.2636324.

[5] Y. Zeng, M. Zhang, F. Han, Y. Gong, and J. Zhang, "Spectrum analysis and convolutional neural network for automatic modulation recognition", *IEEE Wireless Communications Letters*, vol. 8, no. 3, pp. 929–932, 2019. DOI: 10.1109/LWC.2019.2900247.

[6] S. Wei, Q. Qu, X. Zeng, J. Liang, J. Shi, and X. Zhang, "Self-attention Bi-LSTM networks for radar signal modulation recognition", *IEEE Transactions on Microwave Theory and Techniques*, vol. 69, no. 11, pp. 5160–5172, 2021. DOI: 10.1109/TMTT.2021.3112199.

[7] E. E. Azzouz and A. K. Nandi, "Modulation recognition using artificial neural networks", in *Automatic Modulation Recognition of Communication Signals*. Springer, Boston, MA, 1996, pp. 132–176. DOI: 10.1007/978-1-4757-2469-1_5.

[8] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks", in *Engineering Applications of Neural Networks. EANN 2016. Communications in Computer and Information Science*, vol. 629. Springer, Cham, 2016, pp. 213–226. DOI: 10.1007/978-3-319-44188-7_16.

[9] Z. Zhang, H. Luo, C. Wang, C. Gan, and Y. Xiang, "Automatic modulation classification using CNN-LSTM based dual-stream structure", *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 13521–13531, 2020. DOI: 10.1109/TVT.2020.3030018.

[10] S. Yang, C. Yang, D. Feng, X. Hao, and M. Wang, "One-dimensional deep attention convolution network (ODACN) for signals classification", *IEEE Access*, vol. 8, pp. 2804–2812, 2020. DOI: 10.1109/ACCESS.2019.2958131.

[11] Y. Shi, H. Xu, L. Jiang, and Z. Qi, "ConvLSTMAE: A spatiotemporal parallel autoencoders for automatic modulation classification", *IEEE Communications Letters*, vol. 26, no. 8, pp. 1804–1808, 2022. DOI: 10.1109/LCOMM.2022.3179003.

[12] 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", in *Proc. of 2020 IEEE 91st Vehicular Technology Conference*, 2020, pp. 1–6. DOI: 10.1109/VTC2020-Spring48590.2020.9128408.

[13] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, and S. Pollin, "Deep learning models for wireless signal classification with distributed low-cost spectrum sensors", *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433–445, 2018. DOI: 10.1109/TCCN.2018.2835460.

[14] H. Chen, W. Guo, K. Kang, and G. Hu, "Automatic modulation recognition method based on phase transformation and deep residual shrinkage network", *Electronics*, vol. 13, no. 11, p. 2141, 2024. DOI: 10.3390/electronics13112141.

[15] F. Zhang, C. Luo, J. Xu, and Y. Luo, "An efficient deep learning model for automatic modulation recognition based on parameter estimation and transformation", *IEEE Communications Letters*, vol. 25, no. 10, pp. 3287–3290, 2021. DOI: 10.1109/LCOMM.2021.3102656.

[16] F. Zhang, C. Luo, J. Xu, and Y. Luo, "An autoencoder-based I/Q channel interaction enhancement method for automatic modulation recognition", *IEEE Transactions on Vehicular Technology*, vol. 72, no. 7, pp. 9620–9625, 2023. DOI: 10.1109/TVT.2023.3248625.

[17] Y. Shi, H. Xu, Y. Zhang, Z. Qi, and D. Wang, "GAF-MAE: A self-supervised automatic modulation classification method based on Gramian Angular Field and Masked Autoencoder", *IEEE Transactions on Cognitive Communications and Networking*, vol. 10, no. 1, pp. 94–106, 2024. DOI: 10.1109/TCCN.2023.3318414.

[18] J. D. Ruikar, D.-H. Park, S.-Y. Kwon, and H.-N. Kim, "HCTC: Hybrid convolutional transformer classifier for automatic modulation recognition", *Electronics*, vol. 13, no. 19, p. 3969, 2024. DOI: 10.3390/electronics13193969.

[19] M. Li, O. Li, G. Liu, and C. Zhang, "Generative adversarial networks-based semi-supervised automatic modulation recognition for cognitive radio networks", *Sensors*, vol. 18, no. 11, p. 3913, 2018. DOI: 10.3390/s18113913.

[20] 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. DOI: 10.1109/TVT.2019.2900460.

[21] A. Tailor, M. Dua, and P. Verma, "Automatic classification of multi-carrier modulation signal using STFT spectrogram and deep CNN", *Physica Scripta*, vol. 99, no. 7, art. 076009, 2024. DOI: 10.1088/1402-4896/ad538a.

[22] 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, 2022. DOI: 10.26599/TST.2020.9010057.

[23] S. Sun and Y. Wang, "A novel deep learning automatic modulation classifier with fusion of multichannel information using GRU", *EURASIP Journal on Wireless Communications and Networking*, vol. 2023, art. no. 66, 2023. DOI: 10.1186/s13638-023-02275-y.

[24] J. Xu, C. Luo, G. Parr, and Y. Luo, "A spatiotemporal multi-channel learning framework for automatic modulation recognition", *IEEE Wireless Communications Letters*, vol. 9, no. 10, pp. 1629–1632, 2020. DOI: 10.1109/LWC.2020.2999453.

[25] N. K. Pathak and V. Bajaj, "Automatic modulation classification using bimodal parallel multichannel deep learning framework for spatial multiplexing MIMO system", *Physical Communication*, vol. 59, art. 102071, 2023. DOI: 10.1016/j.phycom.2023.102071.

[26] Y. Li, X. Shi, H. Tan, Z. Zhang, X. Yang, and F. Zhou, "Multi-representation domain attentive contrastive learning based unsupervised automatic modulation recognition", *Nature Communications*, vol. 16, art. no. 5951, pp. 1–13, 2025. DOI: 10.1038/s41467-025-60921-z.