

Article

An Improved Modulation Recognition Algorithm Based on Fine-Tuning and Feature Re-Extraction

Yibing Wang ¹, Liang Zhou ², Zhutian Yang ^{1,*}, Longwen Wu ¹ , Zhendong Yin ¹, Yaqin Zhao ¹  and Zhilu Wu ¹

¹ School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China; 22b905058@stu.hit.edu.cn (Y.W.); wulongwen@hit.edu.cn (L.W.); yinzhendong@hit.edu.cn (Z.Y.); yaqinzhao@hit.edu.cn (Y.Z.); wuzhilu@hit.edu.cn (Z.W.)

² School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China; zlzl@uestc.edu.cn

* Correspondence: yangzhutian@hit.edu.cn

Abstract: Modulation recognition is an important technology in wireless communication systems. In recent years, deep learning-based modulation recognition algorithms, which can autonomously learn deep features and achieve superior recognition performance compared with traditional algorithms, have emerged. Yet, there are still certain limitations. In this paper, aiming at addressing the issue of poor recognition performance at low signal-to-noise ratios (SNRs) and the inability of deep features to effectively distinguish among all modulation types, we propose an optimization scheme for modulation recognition based on fine-tuning and feature re-extraction. In the proposed scheme, the network is firstly trained with the signals at high SNRs; then, the trained network is fine-tuned to the untrained network at low SNRs. Finally, on the basis of the features learned by the network, deeper features with enhanced discriminability for confused modulation types are obtained using feature re-extraction. The simulation results demonstrate that the proposed optimization scheme can maximize the performance of the neural network in the recognition of signals that are easily confused and at low SNRs. Notably, the average recognition accuracy of the proposed scheme was 91.28% within an SNR range of -8 dB to 18 dB, which is an improvement of 8% to 17% in comparison with four existing schemes.



Citation: Wang, Y.; Zhou, L.; Yang, Z.; Wu, L.; Yin, Z.; Zhao, Y.; Wu, Z. An Improved Modulation Recognition Algorithm Based on Fine-Tuning and Feature Re-Extraction. *Electronics* **2023**, *12*, 2134. <https://doi.org/10.3390/electronics12092134>

Academic Editor: Athanasios D. Panagopoulos

Received: 22 March 2023

Revised: 21 April 2023

Accepted: 4 May 2023

Published: 6 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: modulation recognition; SNR region classification; transfer learning; feature re-extraction

1. Introduction

The incessant advancement in communication technology has profoundly impacted various aspects of social life, and the demand for wireless communication continues to escalate. Typically, signals undergo appropriate modulation during transmission, and as the transmission environment grows increasingly complex, multiple modulation types are included within the communication frequency band [1]. Consequently, it is important to investigate modulation recognition techniques for communication signals in depth. In non-cooperative communication systems [2,3], modulation recognition primarily serves to process the received signals; analyze the modulation type; and subsequently perform signal demodulation, decoding and other operations to obtain valuable information. In cooperative communication systems, modulation recognition techniques are also applied in numerous fields, including spectrum sensing [4,5], spectrum resource management [6], cognitive radio [7] and others. In summary, to guarantee communication security, relevant departments must reinforce the supervision of communication signals. This requires the effective identification of interference information embedded within signals, and modulation recognition can play a crucial role in achieving the efficient allocation of spectrum resources.

Most current modulation recognition techniques are based on likelihood ratio theory or feature extraction algorithms, which involve intricate steps and exacting conditions. A primary drawback of these approaches is that feature extraction and selection may result

in the loss of some signal information. Consequently, neural network-based modulation recognition algorithms have garnered attention, as they can achieve end-to-end recognition without manual feature extraction. This class of algorithms can retain the signal information to the maximum extent and achieve better results. Neural network-based modulation recognition techniques for communication signals are more suitable for the emerging modulation types. However, the deep features extracted by neural networks cannot effectively recognize all modulation types, resulting in confusion among certain modulation types. The existing methods attempt to resolve this issue by increasing the number of network layers, such as implementing deep neural networks such as residual network 50 (ResNet50), to improve the modulation recognition rates. Nonetheless, when the dataset is large and the network has numerous parameters to learn, it takes a long time to train the network, thereby diminishing model efficiency. Furthermore, when the neural network is initialized with random weights and trained several times on the same dataset, the network recognition performance in each training process considerably varies. The modulation recognition rates of the same network trained with signals at high SNRs and low SNRs also exhibit significant disparities.

In this paper, we focus on the problem that existing neural network-based modulation recognition algorithms achieve poor recognition of signals with noise interference or easily confused. To enhance the recognition performance, a method based on fine-tuning and feature re-extraction is proposed to effectively recognize 11 modulation types in the dataset RadioML2016.10a (RML2016.10a), i.e., 16-ary quadrature amplitude modulation (16QAM), 64-ary quadrature amplitude modulation (64QAM), binary phase-shift keying (BPSK), quaternary phase-shift keying (QPSK), eight-level phase-shift keying (8PSK), continuous-phase frequency-shift keying (CPFSK), Gaussian frequency-shift keying (GFSK), four-level pulse-amplitude modulation (PAM4), amplitude modulation single sideband (AM-SSB), amplitude modulation double sideband (AM-DSB) and wideband frequency modulation (WBFM). Firstly, the dataset is divided into several subregions according to the SNR values. Secondly, the modulated signals at high SNRs (source data) are used to train the source network. Thirdly, the weights of the trained source network are transferred to the untrained target network as the initial weights. Fourthly, the untrained target network is trained with the signals at low SNRs (target data). Finally, feature re-extraction is performed if the target network has been trained. The main contributions of this paper are outlined as follows:

- (1) A novel modulation recognition algorithm based on fine-tuning and feature re-extraction is proposed, and the proposed algorithm can improve the performance of the neural network in the recognition of the signals that are easily confused at low SNRs.
- (2) With the fine-tuning method, we can transfer the weights of the networks trained with the modulated signals at different SNRs. This can improve the recognition accuracy for signals at low SNRs, as well as the stability of the network.
- (3) Since neural networks cannot achieve good recognition of all modulation types, we propose the feature re-extraction method. With the method, deeper features are extracted from the outputs of the trained network's penultimate layer, thus achieving the effective recognition of easily confused modulation types.
- (4) Finally, the combination of fine-tuning and feature re-extraction can improve recognition performance to the maximum extent.

The simulation results confirm that the proposed algorithm achieved better recognition performance than state-of-the-art modulation recognition algorithms. We further explored the rationality of our proposed algorithm with controlled groups of experiments and analyzed the aspects of the confusion matrix and model complexity.

The rest of this paper is organized as follows: We review related literature in Section 2 and present the system model in Section 3. Then, we propose a modulation recognition algorithm combining fine-tuning and feature re-extraction and discuss the algorithm design process in Section 4. Simulation results and performance evaluation are provided in Section 5. Finally, the conclusion is drawn in Section 6.

2. Literature Review of Related Works

In communication systems, a baseband signal needs to be modulated for transmission in the channel. With the development of communication technology, there are various modulation types with different characteristics. Modulation recognition is a two-step process: pre-processing the communication signals and using the appropriate classifier to recognize the modulation types [8]. The modulation recognition algorithms for communication signals can be divided into three categories at present [9], which are likelihood-based, feature-based and deep learning-based algorithms.

The modulation recognition algorithm based on the likelihood function, which successfully distinguishes between BPSK and QPSK signals, was firstly proposed in [10]. More specifically, the authors calculated the probability density functions of signal parameters, such as the symbol transmission rate, the SNR and the carrier frequency; obtained the corresponding log-likelihood ratio; and then estimated the modulation order of the signals. However, the derivation process of the likelihood function is computationally complex and requires a priori knowledge about the distribution of statistics [11]. Moreover, the specific decision criteria for the likelihood ratio are also different for different practical problems, so likelihood-based modulation recognition algorithms are less generalizable. In addition, it is difficult to obtain accurate values of signal parameters at low SNRs, which affects the recognition of the modulation types.

The modulation recognition algorithm based on signal feature extraction [12,13] consists of the following three steps: Firstly, we should pre-process the modulated signals, mainly including signal down-sampling, digital filtering, etc. Secondly, we can extract the features from different angles to realize effective signal description. Finally, based on the differences among the corresponding signal eigenvalues, we can recognize the modulated signals by setting appropriate thresholds. Zhang et al. [14] constructed six characteristic parameters based on instantaneous information and signal spectrum. The proposed method correctly classified the modulated signals of two-level amplitude-shift keying (2ASK), four-level amplitude-shift keying (4ASK), two-level frequency-shift keying (2FSK), BPSK, minimum shift keying (MSK), frequency modulation (FM), lower sideband (LSB) and upper sideband (USB) with more than 95% recognition rate at SNR = 6 dB. On the basis of high-order cumulants, combined with peak features of the FFT spectrum and instantaneous signal features, Yang et al. [15] proposed a new method for digital modulation recognition based on mixed signal features. The new method successfully and efficiently recognized six classical digital modulation types and achieved satisfactory recognition results even at rather low SNRs. By considering the different cumulant combinations of 2FSK, 4FSK, BPSK, QPSK, 2ASK and 4ASK signals, Xie et al. [16] established new signal parameters to achieve better recognition of these digital modulation types. The overall recognition accuracy was 99% at SNR = -5 dB and 100% at SNR = -2 dB. Wang et al. [17] used the fourth-order cumulants of four signals (8PSK, 16QAM, PAM4 and BPSK) as the recognition parameters. Under additive white Gaussian noise (AWGN) channels, the recognition accuracy reached more than 90% when the number of symbols was above 250 and SNR > 10 dB. Hassanpour et al. [18] proposed a wavelet-based algorithm for the recognition of binary digital modulation types, including 2ASK, 2FSK and BPSK, in the presence of AWGN. The average rates of 99.97%, 99.71% and 97.34% were obtained for the recognition of the three modulations at -5 dB, -7 dB and -10 dB. Yang et al. [19] converted the time-domain diagrams of different complex modulated signals into spectrogram images using the wavelet transform. Then, the authors adopted AlexNet to classify the eight modulated signals of 2ASK, 4ASK, 2PSK, 4PSK, 2FSK, 4FSK, 16QAM and 64QAM. The recognition accuracy of the eight modulation types was almost 100% at higher SNRs. In [20], a new blind modulation classification (BMC) method was proposed for classifying the three modulated signals of QPSK, offset-QPSK (OQPSK) and $\pi/4$ -QPSK, based on the second-order and fourth-order cyclic cumulants. The proposed feature-based BMC algorithm added robustness against various impairments and worked well even in the frequency-selective fading channels. Wei et al. [21] proposed a novel method for the automatic modulation classification (AMC) of digital

communication signals using a support vector machine (SVM) based on hybrid features, cyclostationarity and information entropy. Moreover, the authors proposed three new features, which did not require any prior information and had a strong anti-noise ability. Shi [22] extracted Box fractal dimension, Katz fractal dimension, Higuchi fractal dimension, Petrosian fractal dimension and Sevcik fractal dimension from eight modulated signals. In addition, back-propagation (BP) neural network, gray relation analysis (GRA), random forest (RF) and K-nearest neighbor (KNN) were used to recognize the different modulated signals based on the fractal features. The results indicated that RF had better recognition performance with 96% accuracy at SNR = 10 dB. Wang et al. [23] proposed a low-complexity graphic constellation projection (GCP) algorithm for AMC, and adopted the deep belief network (DBN) to learn the underlying features in these constellations. The recognition accuracy was beyond 95% at SNR = 0 dB. Yan et al. [24] presented an innovative AMC method using graph-based constellation analysis for M -ary QAM signals. The proposed method with lower computational complexity could provide superior performance compared with existing subtractive clustering techniques and was robust to the residual phase and timing offsets. In summary, modulation recognition performance can be improved by extracting features with significant differences among the modulation types from multiple perspectives. Moreover, it is necessary to select an appropriate classifier in order to obtain better recognition performance. The feature-based modulation recognition algorithm is less computationally intensive and simpler to implement than the likelihood-based one, but the recognition performance depends on the number of features and the differences among features. Moreover, it is difficult to accurately extract features in non-ideal channels.

In recent years, with the rapid development of deep learning, researchers have started applying it to signal processing [25–30]. The main innovation point of deep learning-based methods is that the novel network architectures with tens or even hundreds of layers and network training methods are allowed to be used for recognition. On the one hand, the deep learning-based modulation recognition algorithm can extract artificial features from the original signals and then utilize the extracted features as the inputs of neural networks. Lee et al. [31] proposed an enhanced blind modulation classification (BMC) method based on deep neural network (DNN) for fading channels. Then, the authors adopted DNN to recognize 16QAM, 64QAM, BPSK, QPSK and 8PSK based on 28 signal features. The experimental results showed that the recognition rate was enhanced with the increase in the number of signal features. Kim et al. [32] adopted deep connected neural network (DCNN) with artificial features as the network inputs to successfully recognize PSK and QAM signals with different orders. The authors discussed the interference of Gaussian white noise and Doppler frequency shift with the network recognition performance and confirmed that DCNN had stronger generalization ability and signal recognition ability. Mendis et al. [33] proposed an automatic modulation classification (AMC) method based on a spectral correlation function (SCF) pattern. The authors used DBN to abstract the complex signal features that were represented by the associated SCF patterns and then distinguished among five kinds of digitally modulated signals using the features. The proposed method had low sensitivity to Gaussian white noise channels. In addition, the recognition accuracy was greatly reduced in the AWGN environment. To solve the problem, a multi-carrier recognition system based on CNN and principal component analysis (PCA) was proposed in [34]. The PCA-based processing method could suppress AWGN and reduce the dimension of the network inputs. The system correctly identified three kinds of multi-carrier waveforms in a dense transmission environment and achieved good recognition results even at low SNRs. Gou et al. [35] proposed a semi-supervised learning method based on data-driven models that combined contrastive predictive coding with an unsupervised pre-training algorithm, as well as a supervised learning algorithm. The authors constructed a joint DNN composed by long short-term memory (LSTM) and ResNet50 and then extracted the instantaneous features using the Hilbert transform as the network inputs to recognize 11 modulation types. The semi-supervised joint neural network structure improved the recognition accuracy by 3~20% compared with the previous

methods and reached an average recognition accuracy of 94% at SNR levels ranging from 0 dB to 18 dB.

On the other hand, deep learning-based recognition algorithms can directly utilize the original signals as the network inputs and realize end-to-end recognition. This class of algorithms have strong generalization ability and robustness for various modulation recognition tasks. O'Shea et al. [36] developed a new end-to-end modulation recognition algorithm based on deep residual network (DRN). The proposed algorithm was feasible in realistic communication environments and achieved higher recognition accuracy at low SNRs than the other methods mentioned in the paper. Zhang et al. [37] used DBN and temporal in-phase and quadrature (IQ) data representation to identify 11 modulation types. The method obtained high recognition accuracy at high SNRs. Vanhoy et al. [38] proposed a branch convolutional neural network (B-CNN) to recognize more than 20 modulated signals. Xu et al. [39] proposed an effective multi-stream network structure, namely, multi-channel convolutional long short-term deep neural network (MCLDNN). The network structure utilized the information of I-channel data, Q-channel data and I/Q-multi-channel data of the original signals and integrated one-dimensional (1D) convolutional, two-dimensional (2D) convolutional and LSTM layers to extract spatio-temporal features. MCLDNN performed significantly better than other network structures above -4 dB SNR and reached an average recognition accuracy of 92% at SNR levels ranging from 0 dB to 18 dB, an improvement of 2~10% over the others.

In practical scenarios, it is difficult to construct large-scale well-annotated datasets for all domains of interest, and the recognition model performs weakly in the domain with insufficient data. To address this problem, Bu et al. [40] proposed an adversarial transfer learning architecture (ATLA), incorporating adversarial training and knowledge transfer in a unified way. The proposed ATLA substantially boosted the performance of the target model. More specifically, the target model achieved the recognition accuracy of 82% with half of the training data reduced, and the accuracy was increased by 17.3% with respect to that of supervised learning with one-tenth of training data. In addition, there are generally few labeled samples and large unlabeled samples in realistic communication scenarios. It is almost impossible to implement previously proposed deep learning-based AMC algorithms in this case. Wang et al. [41] proposed a TL-based semi-supervised AMC (TL-AMC) method in a zero-forcing-aided multiple-input and multiple-output (ZF-MIMO) system. TL-AMC performed better than CNN-based AMC with the limited samples, and TL-AMC also achieved recognition accuracy at high SNRs similar to that of CNN-based AMC trained on massive labeled samples. Most of existing AMC methods have been designed under the assumption that the classifier has prior knowledge of the signal and channel parameters. Perenda et al. [42] proposed two possible directions to make AMC more robust to signal shape transformations introduced by unknown signal and channel parameters. Spatial transformer networks (STNs) and TL were embedded into a light ResNeXt (ResNet next dimension)-based classifier. This proposed method improved the average recognition accuracy up to 10~30% in specific unseen scenarios, with only 5% of labeled data for a large dataset of 20 complex higher-order modulation types. Finally, Table 1 presents the summary of the above-mentioned deep learning-based modulation recognition algorithms and compares deep learning-based algorithms and the proposed algorithm in terms of advantages, limitations and recognition accuracy.

With the rapid development of communication technology, the demand for automatic modulation recognition (AMR) in signal processing scenarios has become increasingly urgent. According to the review of the related literature on modulation recognition algorithms, deep learning-based algorithms can automatically extract deep features to achieve AMR, but there are still some problems. Therefore, an improved modulation recognition algorithm based on fine-tuning and feature re-extraction is proposed in this paper.

Table 1. Comparison of deep learning-based algorithms and the proposed algorithm.

Reference	Method	Modulation Set	Advantages	Limitations	Recognition Accuracy
[31]	28 statistical features + DNN	BPSK, QPSK, 16QAM, 64QAM and 8PSK	Good performance and robust to fading channels	Poor recognition of 16QAM and 64QAM	Acc = 86.43% when the Doppler frequency and SNR are set to 100 Hz and 5 dB
[32]	21 statistical features + DNN	BPSK, QPSK, 16QAM, 64QAM and 8PSK	Good discrimination of modulation types for high-Doppler fading channels	Unreliable performance comparison with previous methods (inconsistent network inputs)	Acc = 100% when SNR > 0 dB
[33]	SCF patterns + DBN	4FSK, 16QAM, BPSK, QPSK and OFDM	Robust at low SNRs	Poor recognition of BPSK and QPSK	Acc > 90% when SNR > -2 dB
[34]	Instantaneous amplitude + CNN	OFDM-QAM, FBMC-OQAM and UFMC	Good discrimination of multicarrier waveforms in dense transmission environment; low computational complexity	Raw amplitude features are not effective at low SNRs	Acc = 97.4% at SNRs ranging from -5 dB to 20 dB
[35]	Instantaneous parameters + LSTM-ResNet	RML2016.10a	Good recognition of modulation types under small-sample-scale conditions	High model complexity; poor recognition of 8PSK and QPSK, 16QAM and 64QAM, and WBFM and AM-DSB	Acc = 92% at 0 dB
[36]	IQ data + ResNet	RML2018.01a	Good performance on the difficult signal database	Poor recognition of high-order modulation types (16-/32-PSK and 64-/128-/256-QAM) and AM modes; requires high SNRs; requires many samples	Acc = 80% at 10 dB
[37]	IQ data + DBN	RML2016.10a	Simple model structure	Poor recognition of 16QAM and 64QAM; requires high SNRs	Acc = 92.12% at 18 dB
[38]	IQ data + CLDNN with hierarchical structure	MGFSK, MCPFSK, MPAM, MQAM, AM, WBFM, IFM, OFDM, MASK and MPSK	Different network structures for different families of modulations	Poor recognition of high-order modulations; recognition rate is enhanced with the increase in sample size	Acc > 75% when SNR > 0 dB; maximum rate is 80%
[39]	IQ data + MCLDNN	RML2016.10a	Efficient convergence speed	Poor recognition of 16QAM and 64QAM, and WBFM and AM-DSB	Acc = 92% at SNRs ranging from 0 dB to 18 dB
[40]	IQ data + ATLA	RML2016.10a	Good recognition performance with insufficient data and under various imperfections (frequency offset, etc.)	Poor recognition of 16QAM and 64QAM, and WBFM and AM-DSB	Acc = 82% with half of the training data; Acc increased by 17.3% with one-tenth of training data
[41]	IQ data + TL-AMC	BPSK, QPSK, 8PSK and 16QAM	Good performance at high SNRs	Poor recognition of 16QAM and 8PSK at low SNRs	Acc = 100% at 10 dB
[42]	IQ data + ResNeXt	RML2016.10a + RML2018.01a	Robust to signal shape transformations introduced by unknown signal and channel parameters	Vulnerable to out-of-distribution data	Acc increased by 10~30% with only 5% of labeled data
Proposed algorithm	Fine-tuned MCLDNN + Feature re-extraction	RML2016.10a	Better recognition of modulation types that are at low SNRs and easily confused	Comparatively high network complexity	Acc = 96.65% at 0 dB; Acc = 91.28% at SNRs ranging from -8 dB to 18 dB

3. System Model

3.1. Signal Model

This paper considers a single-input single-output communication system, and the received signal, $r(t)$, can be represented by

$$r(t) = s(t) * h(t) + n(t), \tag{1}$$

where $s(t)$ is the modulated signal from the transmitter, $h(t)$ is the channel impulse response, $n(t)$ denotes AWGN and $*$ represents the convolution operation. The received signal, $r(t)$, is sampled n times at a sampling rate $f_s = 1/T_s$ (sampling period) by the analog-to-digital converter, which generates the discrete-time observed signal, $r(n)$.

3.2. Network Model

Communication signals contain both spatial characteristics and temporal correlations, so MCLDNN [39], which integrates CNNs, LSTMs and fully connected (FC) deep neural networks in a unified structure, can utilize their synergy for spatiotemporal feature extraction. Moreover, the imbalance between signal amplitude and phase deteriorates the orthogonality between the I channel and the Q channel and leads to an inherent difference between the two channels. So, the features extracted from the I-channel, the Q-channel and the I/Q-multi-channel data are complementary.

MCLDNN comprises four distinct functional parts: multi-channel inputs, spatial characteristic mapping, temporal characteristic extraction and fully connected classification (local feature integration). The framework is shown in Figure 1. Specifically, the corresponding convolution operations are firstly performed on the modulated signals of each input channel. Then, the multiple feature maps obtained after convolution are spliced and fused using the concatenate layer. Finally, the fused features are transmitted to the LSTM layer to further extract the temporal correlation features and classify the signals.

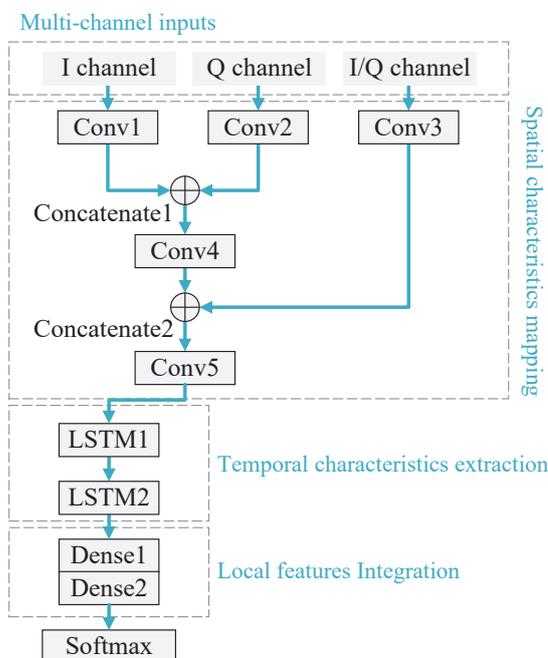


Figure 1. MCLDNN network structure diagram.

Signal modulation is essentially a process of converting the amplitude, phase and frequency of signals according to specific laws. The backbone neural network adopted in this paper adds instantaneous frequency and instantaneous phase as the network inputs on the basis of MCLDNN, making it theoretically applicable to various modulation types.

3.3. Recognition Methods

The improved scheme proposed in this paper focuses on the deep learning-based modulation recognition of signals transmitted in single-user noisy channels in the non-cooperative communication scenario. In this scenario, the neural networks with randomly initialized weights have large differences in the recognition of signals at high SNRs and low SNRs. Thus, we firstly use the second-order and fourth-order moment (M_2M_4) algorithm to divide the received signals into three categories according to the SNR values. Then, we transfer the network weights based on the fine-tuning method, thereby improving the network recognition performance of noisy signals. Furthermore, the transmitter can control the data rate and signal bandwidth using signal modulation; then, the receiver may not identify some modulated signals with similar attributes. To solve the problem, the feature re-extraction method is proposed to obtain the more discriminative features of easily confused signals and achieve effective recognition.

3.3.1. SNR Estimation Based on the M_2M_4 Algorithm

SNR values are important indicators to measure channel quality. However, in realistic communication scenarios, the receiver does not have any known information about the received signals, so it is necessary to estimate the SNR values of the signals. D. R. Pauluzzi [43] sketched the derivation of the M_2M_4 estimator for complex channels and then showed how the estimator could be modified for application to real channels using the same approach. Let M_2 and M_4 denote the second-order and fourth-order moments of the sampled data y_n :

$$M_2 = E\{y_n y_n^*\} = E\{a_n a_n^* + w_n w_n^* + a_n w_n^* + a_n^* w_n\}, \tag{2a}$$

$$M_4 = E\{(y_n y_n^*)^2\} = E\{(a_n a_n^*)^2 + (w_n w_n^*)^2 + (a_n w_n^*)^2 + (a_n^* w_n)^2 + 4(a_n a_n^* w_n w_n^*) + 2(a_n a_n^* a_n w_n^*) + 2(a_n a_n^* a_n^* w_n) + 2(w_n w_n^* a_n w_n^*) + 2(w_n w_n^* a_n^* w_n)\}. \tag{2b}$$

where a_n is the signal constituent of y_n and w_n is the noise constituent of y_n .

Assuming that the signal and noise are zero-mean, independent random processes, and that the in-phase and quadrature components of the noise are independent, (2a) and (2b) are written as

$$M_2 = E\{a_n a_n^*\} + E\{w_n w_n^*\}, \tag{3a}$$

$$M_4 = E\{(a_n a_n^*)^2 + (w_n w_n^*)^2 + 4(a_n a_n^* w_n w_n^*)\} = E\{(a_n a_n^*)^2\} + E\{(w_n w_n^*)^2\} + 4E\{a_n a_n^*\}E\{w_n w_n^*\}, \tag{3b}$$

and for the sake of a simple notation, the following abbreviations are introduced:

$$S := E\{a_n a_n^*\}, \tag{4a}$$

$$N := E\{w_n w_n^*\}, \tag{4b}$$

where S is the average energy of a_n and N is the average energy of w_n . Therefore, (3a) and (3b) can be written as

$$M_2 = S + N, \tag{5a}$$

$$M_4 = k_a S^2 + k_w N^2 + 4SN, \tag{5b}$$

where $k_a = E\{|a_n|^4\}/E\{|a_n|^2\}^2$ and $k_w = E\{|w_n|^4\}/E\{|w_n|^2\}^2$ are the kurtosis of the signal and the kurtosis of the noise, respectively. By solving for S and N , one obtains

$$\begin{cases} N = M_2 - S, \\ S = \frac{M_2(k_w - 2) \pm \sqrt{(4 - k_a k_w)M_2^2 + M_4(k_a + k_w - 4)}}{k_a + k_w - 4}, \end{cases} \quad (6a)$$

and the estimator formed as the ratio of S to N is denoted as the M_2M_4 estimator. As an example, for any M -ary PSK signal, $k_a = 1$, and for complex noise, $k_w = 2$, so that

$$\rho_{M_2M_4,complex} = \frac{\sqrt{2M_2^2 - M_4}}{M_2 - \sqrt{2M_2^2 - M_4}}. \quad (7)$$

In a similar manner, assuming that y_n is real, $M_2 = E\{y_n^2\}$ is equivalent to (5a), but $M_4 = E\{y_n^4\}$ is given by

$$M_4 = k_a S^2 + k_w N^2 + 6SN, \quad (8)$$

solving (5a) and (8) for N gives the same expression as (6a), but the solution for S is

$$S = \frac{M_2(k_w - 3) \pm \sqrt{(9 - k_a k_w)M_2^2 + M_4(k_a + k_w - 6)}}{k_a + k_w - 6}, \quad (9)$$

as an example, for BPSK signals, $k_a = 1$, and for real noise, $k_w = 3$, so that

$$\rho_{M_2M_4,real} = \frac{0.5\sqrt{6M_2^2 - 2M_4}}{M_2 - 0.5\sqrt{6M_2^2 - 2M_4}}. \quad (10)$$

In practice, the second-order and fourth-order moments are estimated using their respective time averages for both real and complex channels as

$$M_2 \approx \frac{1}{M} \sum_{n=1}^M |y_n|^2, \quad (11a)$$

$$M_4 \approx \frac{1}{M} \sum_{n=1}^M |y_n|^4, \quad (11b)$$

where M denotes the number of the floating-point time I/Q samples for each signal datum.

The M_2M_4 algorithm [44] has low computational complexity, and it is insensitive to carrier deviation and phase deviation. Moreover, since the algorithm allows blind estimation to be conducted, it is widely applied in practice. Related studies have shown that as the number of samples increases, the estimates are closer to the true values. In addition, the algorithm can obtain a desired SNR estimator by introducing the combination of higher-order moments according to the actual situation and performance requirements.

3.3.2. Fine-Tuning

Transfer learning is to transfer the knowledge learned from the source domain to the target domain. It includes two important concepts: domain and task [45], where a domain \mathcal{D} is composed of a d -dimensional feature space \mathcal{X} and a marginal probability distribution function $P(X)$, where $X = \{x_1, \dots, x_n\} \in \mathcal{X}$, i.e., $\mathcal{D} = \{\mathcal{X}, P(X)\}$. Given a specific domain \mathcal{D} , a task \mathcal{T} is composed of a label space \mathcal{Y} and a predictive function $f(\cdot)$, i.e., $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$. The function $f(\cdot)$ can be used to predict the corresponding label, $f(x)$, of a new sample x . From a probabilistic viewpoint, $f(x)$ is approximately equal to $P(y|x)$,

that is, the probability distribution of y under the condition of a given x . Based on the above, transfer learning can be defined as follows: Given source domain

$$\mathcal{D}_S = \{(x_1^s, y_1^s), (x_2^s, y_2^s), \dots, (x_{n_s}^s, y_{n_s}^s)\}, \tag{12}$$

and learning task \mathcal{T}_S , and target domain

$$\mathcal{D}_T = \{(x_1^t, y_1^t), (x_2^t, y_2^t), \dots, (x_{n_t}^t, y_{n_t}^t)\}, \tag{13}$$

and learning task \mathcal{T}_T , transfer learning aims to help improve the learning of the target predictive function $f_T(\cdot)$ in \mathcal{D}_T using the knowledge in \mathcal{D}_S and \mathcal{T}_S , where $x_i^s \in \mathcal{X}_S (i = 1, 2, \dots, n_s)$ and $x_j^t \in \mathcal{X}_T (j = 1, 2, \dots, n_t)$ are the data samples from the source and target domain, respectively; $y_i^s \in \mathcal{Y}_S$ and $y_j^t \in \mathcal{Y}_T$ are the class labels corresponding to the source and target domain sample, respectively; and $\mathcal{D}_S \neq \mathcal{D}_T$ or $\mathcal{T}_S \neq \mathcal{T}_T$.

Fine-tuning is a typical transfer learning method that has been widely used in deep neural networks [46,47]. As shown in Figure 2, the central idea is to transfer the weights of the source network to the target network as its initial weights according to the similarities between the source domain and the target domain. Moreover, the network may have some abnormal conditions with a small sample size, such as the inability to converge, low recognition accuracy, over-fitting and poor generalization ability during the actual training process. Fine-tuning can effectively alleviate the above problems, because most of the source networks have been trained on a large number of data, which is equivalent to expanding the target dataset. With this method, the final target network has strong scalability and robustness. In summary, fine-tuning can avoid retraining the new network and save computational resources and training time, as well as improving model performance.

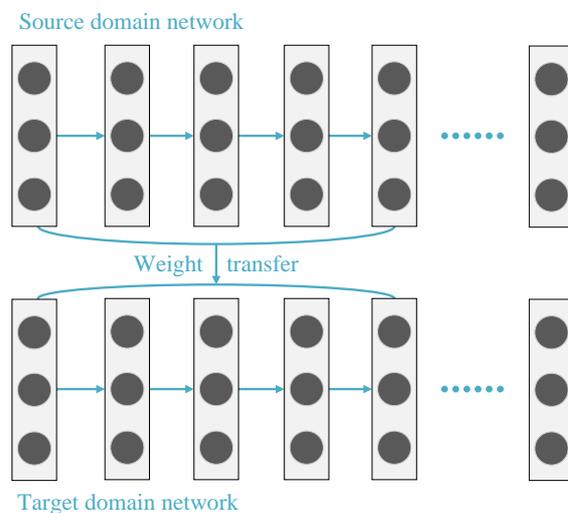


Figure 2. Fine-tuning schematic diagram.

In this paper, we utilize the parameters learned from signals at high SNRs as the initial weights of the backbone neural network at low SNRs. The recognition performance of the modulated signals at low SNRs is effectively improved. The source task and the target task are the same in this case, both of which aim to identify different modulation types. However, the source domain has relatively ideal inputs, while the data in the target domain are more contaminated by noise and interference.

4. Algorithm Design

4.1. Data

GNUradio is an open-source collection of signal processing routines, together with the inception of commercially available software radio front-ends to complete the signal chain. T. J. O’Shea [48] used this software toolkit to generate communication signals; then, the

author used the Hilbert transform on the signals to obtain the transformed signals. The original signals and the corresponding transformed signals were used as the I-channel data and Q-channel data, respectively, and the corresponding SNR value and modulation type of each sample were marked. Finally, multiple modulated signal datasets were generated.

This paper adopted an open-source dataset, RML2016.10a, which includes 220,000 modulated signals with 11 modulation types: BPSK, QPSK, 8PSK, 16QAM, 64QAM, PAM4, CPFSK, GFSK, AM-SSB, AM-DSB and WBFM. The SNR values of the modulated signals vary from -20 dB to +18 dB, at 2 dB intervals. Out of 1000 samples of each modulation type per SNR, 600 samples were randomly selected as training data, 200 samples as validation data and 200 samples as test data. Each sample in the dataset has 128 complex floating-point time I/Q samples and was generated in harsh simulated propagation environments, corrupted by AWGN, multi-path fading, sampling rate offset and center frequency offset to resemble practical environments.

Figures 3–6 display the waveform, instantaneous amplitude, instantaneous frequency and instantaneous phase of one sample of the 11 modulation types. In Figure 3, the blue curve represents the I-channel data, and the red curve represents the Q-channel data. It can be observed in the four figures that the I-channel data, Q-channel data and instantaneous parameters of the 11 modulated signals present large differences. Regarding instantaneous amplitude, BPSK, PAM4, CPFSK, GFSK, AM-SSB, AM-DSB and WBFM are different from each other. However, QPSK is similar to 8PSK, and 16QAM is similar to 64QAM. Regarding instantaneous frequency and phase, the 11 modulation types are different from each other. Therefore, the backbone neural network adds instantaneous frequency and phase as two new input channels on the basis of MCLDNN.

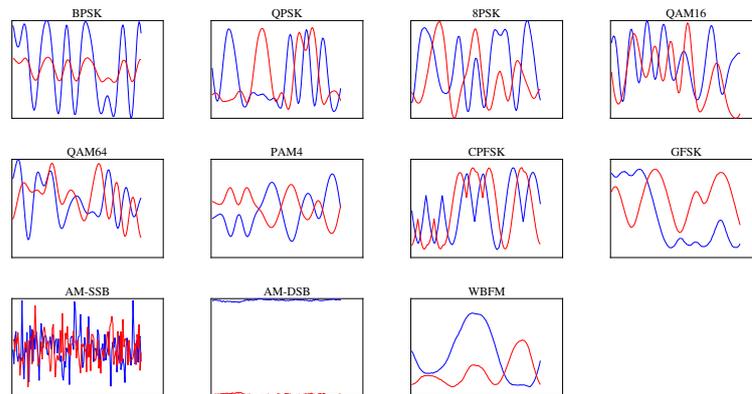


Figure 3. Waveform graphs of 11 modulation types.

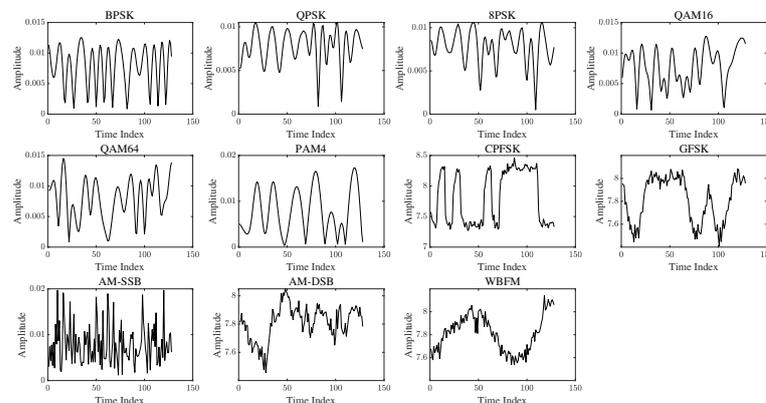


Figure 4. Instantaneous amplitude of 11 modulation types.

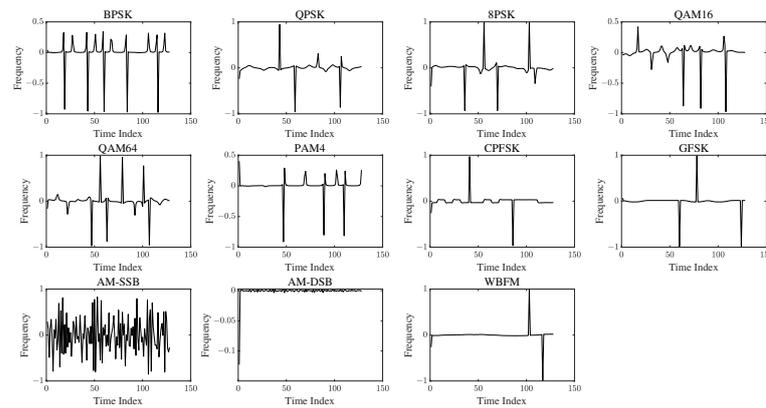


Figure 5. Instantaneous frequency of 11 modulation signals.

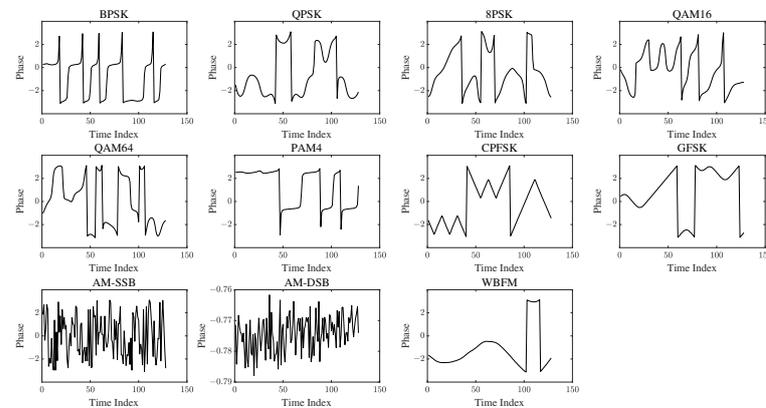


Figure 6. Instantaneous phase of 11 modulation signals.

4.2. Backbone Neural Network

4.2.1. Network Structure

The backbone neural network adds two input channels on the basis of MCLDNN. The network inputs adopt five channels, which are I-channel data $r^I(n)$, Q-channel data $r^Q(n)$, I/Q-multi-channel data $r^{(I,Q)}(n)$, instantaneous frequency $r^F(n)$ and instantaneous phase $r^P(n)$ of the received signal, $r(n)$. The network structure is shown in Figure 7.

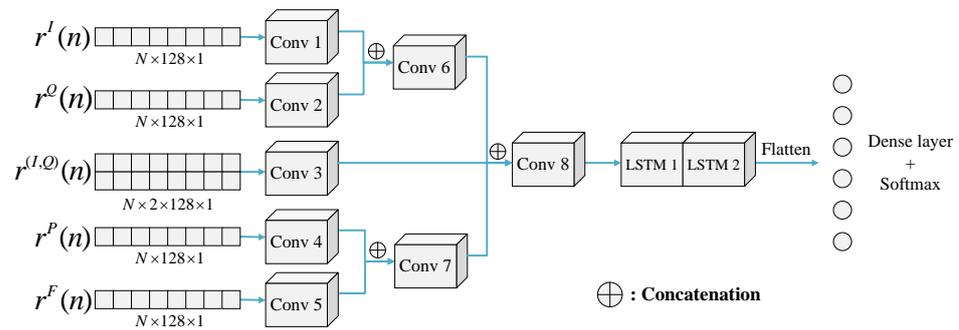


Figure 7. Backbone neural network structure diagram.

In the backbone neural network, Conv1, Conv2, Conv4 and Conv5 are all 1D convolutional layers using 50 convolution kernels with a size of 8, the causal-padding scheme and the Glorot uniform initializer; Conv3 is a 2D convolutional layer using 50 convolution kernels with a size of 2×8 , the same-padding scheme and the Glorot uniform initializer; Conv6 and Conv7 are 2D convolutional layers using 50 convolution kernels with a size of 1×8 , the same-padding scheme and the Glorot uniform initializer; Conv8 is a 2D convolutional layer using 100 convolution kernels with a size of 2×5 , the valid-padding scheme

and the Glorot uniform initializer. These convolutional layers provide superior features to the LSTM layers by reducing noise variance and feeding higher-level abstraction of the input data. Then, both LSTM1 and LSTM2 are LSTM layers with 128 cells to effectively process sequential data and extract temporal correlations of each sample. For mapping features to a more separable space, we added two dense layers with 128 neurons to deepen our network. The output layer uses the softmax function, with 11 neurons corresponding to a modulation mode.

4.2.2. Parameter Optimization

Many hyperparameters need to be tuned to generate a robust neural network that can accurately recognize modulation types. In addition, these hyperparameters can affect the performance of the network, along with its time to convergence. It is difficult to analyze the recognition performance of the neural network using the mathematical derivation, so this section will present the optimal hyperparameters determined with controlled groups of experiments. Specifically, we utilized five modulation types (BPSK, GFSK, AM-SSB, QAM16 and WBFM) in the RML2016.10a dataset as the experimental data to investigate the optimal selection of the network hyperparameters. In this paper, the selection of learning rate and batch size was considered.

The main idea of the BP algorithm [49] is to minimize the cost function by continuously updating the network parameters. This often involves some iterative procedure that applies changes to the parameters at each iteration of the algorithm. We consider the gradient descent algorithm that attempts to optimize the objective function by following the steepest descent direction given by the negative of the gradient. This approach can be applied to update any parameters for which a derivative can be obtained, and the update rule is defined as

$$\theta_{n+1} = \theta_n - \eta \frac{\partial J(\theta_n)}{\partial \theta_n}, \quad (14)$$

where θ_{n+1} and θ_n are the parameter values at the $(n + 1)$ -th iteration and n -th iteration, respectively; $J(\cdot)$ denotes the cost function; $\partial(\cdot)$ means the partial derivatives; and so $\frac{\partial J(\theta_n)}{\partial \theta_n}$ is the gradient of the parameters at the n -th iteration. η is a learning rate that controls how large of a step to take in the direction of the negative gradient.

Setting the learning rate typically involves a tuning procedure in which the optimal learning rate is chosen by hand. Choosing higher than this rate can cause the network to diverge in terms of the objective function, and choosing this rate as too low results in slow learning. In this paper, the optimal learning rate was determined using simulation experiments, and the experimental results are shown in Table 2 and Figure 8.

Table 2. The complexity of the backbone neural network under different learning rates (batch size = 512).

Learning Rate	Training Time (Seconds/Epoch)	Training Epochs	Total Training Time (Seconds)
0.1	299	11	3289
0.05	327	11	3597
0.01	780	28	21,840
0.005	720	24	17,280
0.001	312	43	13,416
0.0005	295	68	20,060
0.0001	358	136	48,688
0.00005	364	198	72,072

Table 2 presents the training time of the backbone neural network at different learning rates, and Figure 8 displays the corresponding recognition accuracy curves. It can be observed that the recognition accuracy reached the highest when the learning rate was in the range of 0.0005~0.001. When the learning rate was greater than 0.001, the recognition

accuracy sharply declined, and the network failed to converge. When the learning rate was less than 0.0005, the recognition accuracy had a slight decrease, but the total training time started to dramatically increase. Therefore, the optimal learning rate was set to 0.001 in this paper.

In addition, the batch size, which is the number of samples used in every epoch to train the network, is also an important hyperparameter. To scale the stochastic gradient-based methods to more processors, it is necessary to increase the batch size to make full use of the computational power of each GPU. However, increasing the batch size often leads to significant loss in test accuracy. In this paper, the batch sizes were in the range of [16, 32, 64, 128, 256, 512, 1024, 2048]. The corresponding training time and training epochs are presented in Table 3. Figure 9 displays the training loss and validation loss of the network for different batch sizes.

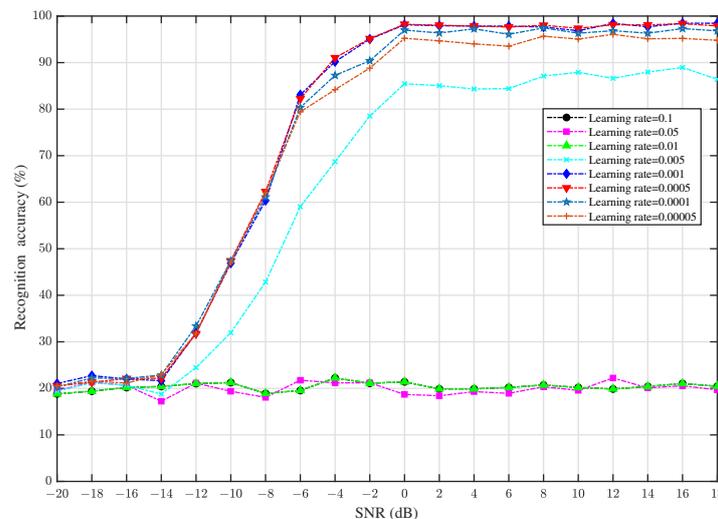


Figure 8. The recognition accuracy of the backbone neural network at different learning rates (batch size = 512).

Table 3. The complexity of the backbone neural network for different batch sizes (learning rate = 0.001).

Batch Size	Training Time (Seconds/Epoch)	Training Epochs	Total Training Time (Seconds)
16	875	24	21,000
32	640	27	17,280
64	455	30	13,650
128	420	38	15,960
256	345	43	14,835
512	312	43	13,416
1024	302	70	21,140
2048	270	70	18,900

It can be seen in Table 3 and Figure 9 that setting the batch size too high made the network take too long to achieve convergence (no more gain in accuracy). However, if the hyperparameter was too low, it made the network bounce back and forth without achieving acceptable performance, and the training time per epoch sharply increased. When the batch size was 512, the total training time was relatively short, and the validation loss curve was relatively smooth, so the optimal batch size of the backbone neural network was set to 512.

According to the above simulation results, the initial learning rate started at 0.001 and multiplied by a factor of 0.8 if the validation loss did not decrease within 5 epochs to improve the training efficiency. The batch size was set to 512 to avoid the local value and speed up the training process. The adaptive moment estimation (Adam) was used in

this paper to minimize the loss function [50], and a dropout rate of $dr = 0.5$ was adopted to avoid overfitting. We stopped the training process when the validation loss did not decrease for 20 epochs and used minimum validation loss to predict the modulation types. All experiments were implemented in the TensorFlow back-end using the Keras deep learning library, supported by a 1 Tesla V100 32 GB GPU.

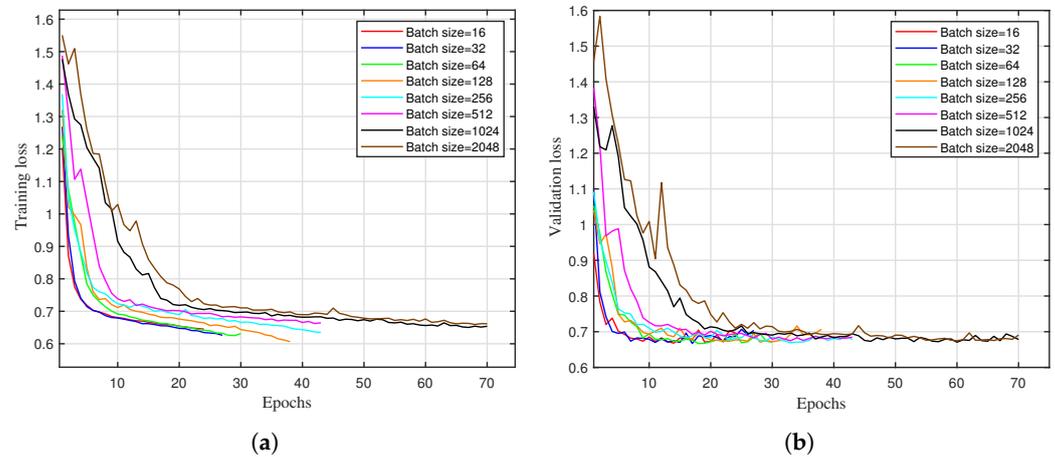


Figure 9. The loss curves of the backbone neural network for different batch sizes (learning rate = 0.001). (a) Training loss and (b) validation loss.

4.3. Signal Feature Extraction

4.3.1. High-Order Cumulant Features

Higher-than-second-order cumulants of Gaussian noise tend to zero, so the noise can be removed from modulated signals using high-order cumulants. In addition, the cumulant values of the signal depend on its modulation type [51,52]. Therefore, high-order cumulants can be used for the recognition of modulated signals with Gaussian white noise. Concretely, let $x(n)$ be a k -th order stationary random process with zero mean, and denote $cum(x_1, x_2, \dots, x_k)$ as the k -th order cumulant of the random vector. The k -th order cumulant of $x(n)$ is defined as

$$C_{kx}(\tau_1, \tau_2, \dots, \tau_{k-1}) = cum(x(n), x(n + \tau_1), \dots, x(n + \tau_{k-1})), \tag{15}$$

the p -th order mixed moment of $x(n)$ is defined as

$$M_{pq} = E\{[x(n)]^{p-q}[x^*(n)]^q\}, \tag{16}$$

where $()^*$ denotes the conjugate operation of a function.

For communication signals, the specific calculation process of high-order cumulants is as follows: The Hilbert transform is firstly used on the received signal, $x(n)$, to obtain the transformed signal, $x_a(n)$. Then, $x(n)$ and $x_a(n)$ are used as the real and imaginary parts to obtain the corresponding analytical signal, $z(n)$, of $x(n)$. Finally, the high-order cumulants of $x(n)$ can be obtained by calculating the mixed moments of $z(n)$. The commonly used higher-order cumulants, $C_{20}, C_{21}, C_{40}, C_{41}, C_{42}, C_{60}, C_{63}$ and C_{80} are defined as follows:

$$C_{20} = cum(x, x) = M_{20}, \tag{17a}$$

$$C_{21} = cum(x, x^*) = M_{21}, \tag{17b}$$

$$C_{40} = cum(x, x, x, x) = M_{40} - 3M_{20}^2, \tag{17c}$$

$$C_{41} = cum(x, x, x, x^*) = M_{41} - 3M_{20}M_{21}, \tag{17d}$$

$$C_{42} = cum(x, x, x^*, x^*) = M_{42} - M_{20}^2 - 2M_{21}^2, \tag{17e}$$

$$C_{60} = cum(x, x, x, x, x, x) = M_{60} - 15M_{20}M_{40} + 30M_{20}^3, \tag{17f}$$

$$C_{63} = cum(x, x, x, x^*, x^*, x^*) = M_{63} - 6M_{41}M_{20} - 9M_{42}M_{21} + 18M_{20}^2M_{21} + 12M_{21}^3, \tag{17g}$$

$$C_{80} = cum(x, x, x, x, x, x, x, x) = M_{80} - 28M_{20}M_{60} - 35M_{40}^2 + 420M_{40}M_{20}^2 - 630M_{20}^4. \tag{17h}$$

In this paper, the following seven features were extracted on the basis of the second-order, fourth-order, sixth-order and eighth-order cumulants of the modulated signals.

$$F_0 = |C_{20}| / |C_{21}|, \tag{18a}$$

$$F_1 = |C_{40}| / |C_{21}|^2, \tag{18b}$$

$$F_2 = |C_{41}| / |C_{21}|^2, \tag{18c}$$

$$F_3 = |C_{42}| / |C_{21}|^2, \tag{18d}$$

$$F_4 = |C_{60}| / |C_{21}|^3, \tag{18e}$$

$$F_5 = |C_{63}| / |C_{21}|^3, \tag{18f}$$

$$F_6 = |C_{20}|^2 / |C_{41}|. \tag{18g}$$

4.3.2. Frequency-Domain Features

Some signals have the same feature values in the time-domain analysis, which requires a further analysis of the signals in other transform domains. In general, the time–frequency transformation of non-periodic dynamic signals can be realized using the Fourier transform, the wavelet transform, the Wigner–Ville distribution and so on [53–55]. The resulting frequency-domain features can distinguish between FM and AM signals.

With the frequency-domain analysis of the difference in the sequence of the instantaneous amplitude of the signals, the characteristic parameter F_7 can be extracted. The calculation process is as follows:

$$\begin{cases} F_7 = \max(|FFT(a_1(n))|), & (19a) \\ a_1(n) = |a(n)| - |a(n - 1)|, & (19b) \end{cases}$$

where $a(n)$ is the amplitude of the sample sequence of the received signal, the corresponding difference is $a_1(n)$ and $FFT(\cdot)$ denotes the fast Fourier transform.

With the frequency-domain analysis of the sample sequence, the characteristic parameter F_8 can be extracted. The calculation process is as follows:

$$\begin{cases} x_1(n) = x(n)^4 - E[x(n)^4], & (20a) \\ s_1(n) = [abs(FFT(x_1(n)))]^2, & (20b) \\ F_8 = \frac{\sqrt{D[s_1(n)]}}{E[s_1(n)]}, & (20c) \end{cases}$$

where $x(n)$ is the sample sequence of the received signal, $E(\cdot)$ denotes the mathematical expectation, $D(\cdot)$ denotes the mean-square deviation and $abs(\cdot)$ is the absolute value.

4.3.3. Spectrum Features

Power spectrum refers to power spectral density (PSD), which can intuitively reflect the power distribution of the modulated signal in frequency. Since the amplitude modulation (AM) signal contains direct current components, its power spectrum has a carrier component, while other signals, such as MPSK, do not have the component at the carrier frequency. Moreover, the number of prominent single-frequency components in the spectral line can be used for the intra-class recognition of MFSK signals [56,57]. On the basis of the cubic, hex and oct spectrums, which are the power spectra of the signal after the operations of cubic, hex and oct powers, the three spectrum features F_9 , F_{10} and F_{11} were obtained in this paper.

We can take F_9 as an example. Firstly, the analytical signal sequence $x_a(n)$ can be obtained using the Hilbert transform on the sample sequence $x(n)$. Then, by using the Fourier transform on the auto correlation function of $x_a(n)$, the estimated value of the signal power spectrum $c_a(n)$ is obtained. Finally, F_9 can be extracted by calculating the standard deviation coefficient of the square of the modulus of $c_a(n)$. The cubic operation is to increase the difference in the spectrum power distribution of the signal, and the standard deviation coefficient is used to measure the degree of variation. The detailed calculation process is as follows:

$$\begin{cases} x_a(n) = H[x(n)^3], & (21a) \\ c_a(n) = FFT\{E[x_a(n)x_a(n + \tau)]\}, & (21b) \\ d_a(n) = |c_a(n)|^2, & (21c) \\ F_9 = \frac{\sqrt{D[d_a(n)]}}{E[d_a(n)]}. & (21d) \end{cases}$$

The calculations of F_{10} and F_{11} are similar to the above process, replacing the cubic power with the sixth and eighth power.

4.3.4. Envelope Features

In an ideal noise-free environment, the envelope of a non-amplitude-modulated signal is generally constant. Although the envelope slightly changes in rare cases, it can still be regarded as constant envelope modulation, while the envelope represents obvious fluctuations for the amplitude-modulated signal. The kurtosis of the normalized-centered instantaneous amplitude [58] can reflect the difference in the amplitude distribution of square QAM signals. The detailed calculation process is defined as

$$\begin{cases} F_{12} = \frac{E[\alpha_{cn}^4(n)]}{E[\alpha_{cn}^2(n)]^2}, & (22a) \\ \alpha_{cn}(n) = \frac{a(n)}{m_\alpha} - 1, & (22b) \\ m_\alpha = \frac{1}{N} \sum_{n=1}^N a(n), & (22c) \end{cases}$$

where $\alpha_{cn}(n)$ is the normalized-centered instantaneous amplitude of the signal and $a(n)$ is the instantaneous amplitude.

4.4. Implementation Details

The flow chart of the modulation recognition algorithm based on fine-tuning and feature re-extraction is shown in Figure 10. The detailed operation steps are as presented below.

(1) Based on the M_2M_4 algorithm, the eigenvalue of each modulated signal in the dataset is calculated. Then, the dataset can be divided into three SNR regions: [12 dB, 18 dB], [0 dB, 6 dB] and [−8 dB, −2 dB].

(2) Based on the fine-tuning method, the weights of network N1 trained with the modulated signals at higher SNRs are used as the initial weights of untrained network N2 at lower SNRs. Then, network N2 is trained with the modulated signals at lower SNRs, and the classification results are obtained.

(3) According to the classification results, we can calculate the recalling rate of each modulation type. If the recalling rate is greater than the threshold, the corresponding modulated signal is classified as the non-confused class, and the classification result is directly outputted. Otherwise, the modulated signal is classified as the confused class.

(4) For the modulated signals belonging to the confused class, on the basis of the deep features extracted by the backbone neural network, the kurtosis of the normalized-

centered instantaneous amplitude is re-extracted to obtain the deeper features with better recognition performance.

(5) For the features obtained in step (4), we use a classifier to identify them and acquire the final classification results.

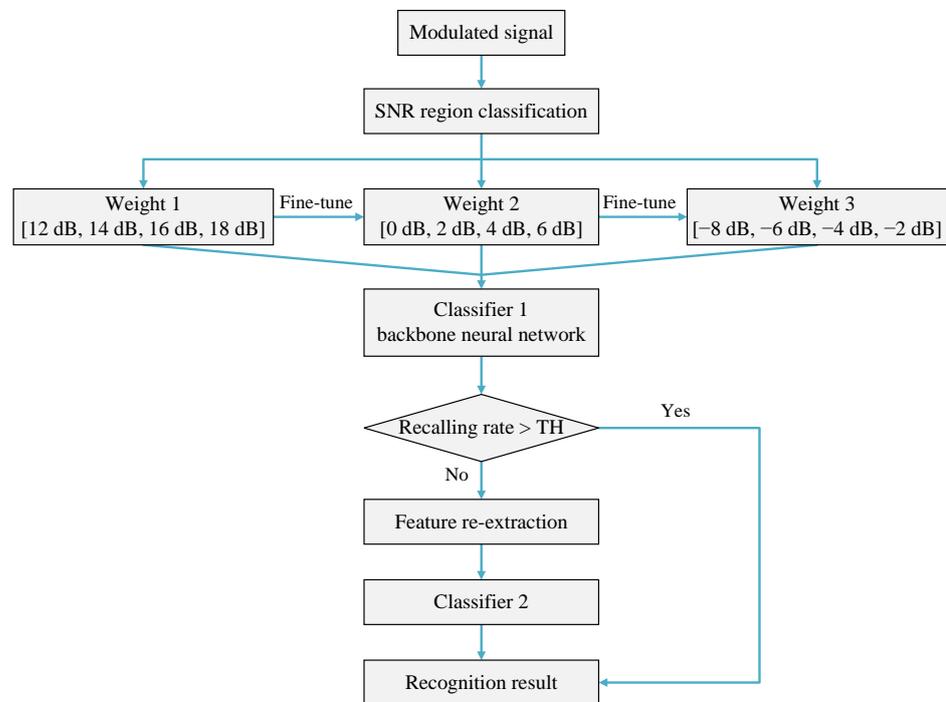


Figure 10. The modulation recognition algorithm flow chart.

5. Simulation Results

5.1. SNR Region Classification

The eigenvalues of all modulated signals calculated by the M_2M_4 algorithm were averaged in segments. Then, by comparing the mean values with the thresholds, the dataset was divided into three SNR subregions. Figure 11 shows the eigenvalue distribution curves of the modulated signals at different SNRs. Figure 11a displays the eigenvalue distribution curves in the SNR region of [12 dB, 18 dB], corresponding to 18 dB, 16 dB, 14 dB and 12 dB, from top to bottom. Figure 11b displays the eigenvalue distribution curves in the SNR region of [0 dB, 6 dB], corresponding to 6 dB, 4 dB, 2 dB and 0 dB from top to bottom. Figure 11c displays the eigenvalue distribution curves in the SNR region of [−8 dB, −2 dB], corresponding to −2 dB, −4 dB, −6 dB and −8 dB from top to bottom.

It can be observed in Figure 11 that the signal eigenvalues in the high-SNR region of [12 dB, 18 dB] were in the range of 200~6000, the signal eigenvalues in the medium-SNR region of [0 dB, 6 dB] were in the range of 5~55 and the signal eigenvalues in the low-SNR region of [−8 dB, −2 dB] were in the range of 2~5. The modulated signal eigenvalues in the three SNR regions could be clearly distinguished, and the classification accuracy was 99.44%.

5.2. Network with Randomly Initialized Weights

Firstly, the backbone neural network for modulation recognition was constructed according to Figure 7. Then, the network with randomly initialized weights was trained with the modulated signals in three SNR regions, and the recognition results of 11 modulation types in each SNR region are shown in Figure 12.

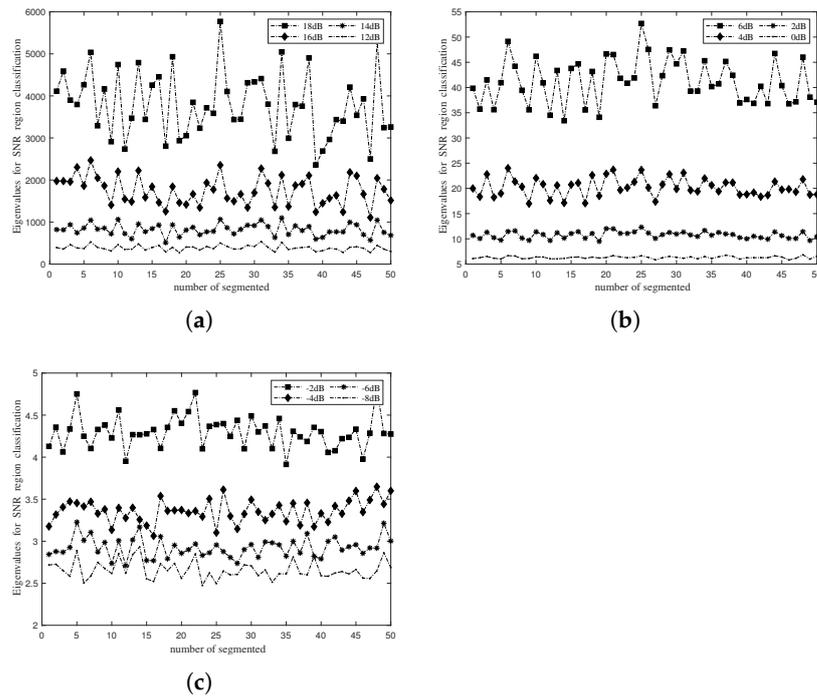


Figure 11. The eigenvalue distribution curves for SNR region classification in (a) [12 dB, 18 dB], (b) [0 dB, 6 dB] and (c) [−8 dB, −2 dB].

It can be seen in Figure 12 that the average recognition accuracy values of 11 modulation types were 91.96%, 82.44% and 54.82% in the three SNR regions of [12 dB, 18 dB], [0 dB, 6 dB] and [−8 dB, −2 dB]. The recognition accuracy shows large differences between the network with randomly initialized weights at high SNRs and that at low SNRs.

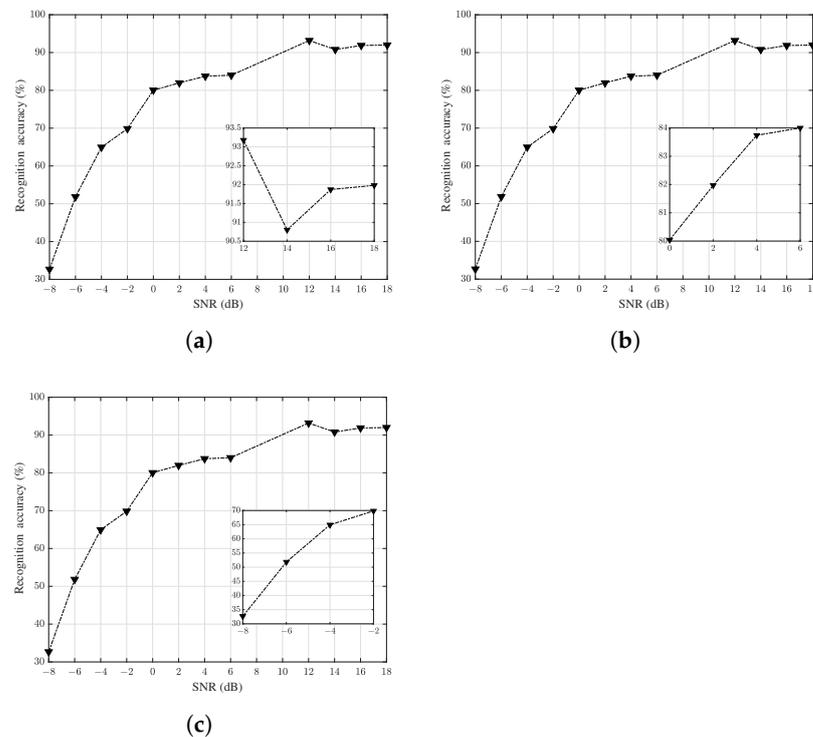


Figure 12. Modulation recognition accuracy of neural networks with randomly initialized weights in the three SNR regions of (a) [12 dB, 18 dB], (b) [0 dB, 6 dB] and (c) [−8 dB, −2 dB].

The confusion matrices for the 11 modulated signals in the high-SNR region of [12 dB, 18 dB] and the medium-SNR region [0 dB, 6 dB] are given in Figure 13. As shown in the figure, BPSK, QPSK, 8PSK, PAM4, CPFSK, GFSK and AM-SSB could be well distinguished by the backbone neural network, except for AM-DSB and WBFM, and 16QAM and 64QAM, which were still confused with each other.

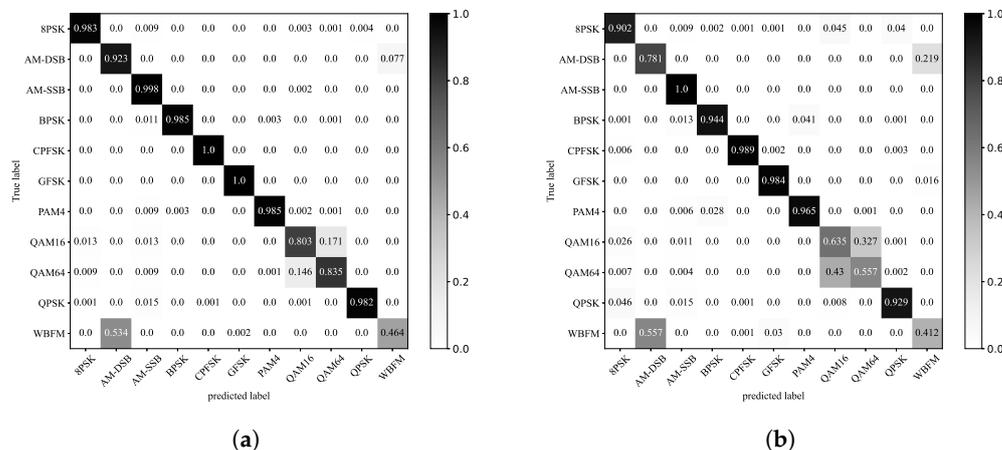


Figure 13. Confusion matrix of the network with randomly initialized weight in the SNR regions of (a) [12 dB, 18 dB] and (b) [0 dB, 6 dB].

5.3. Network with Weight Transfer

The modulated signals in the lower-SNR regions were used to verify the recognition performance of the fine-tuning-based network weight transfer method. The weights of the network trained with the signals in the high-SNR region of [12 dB, 18 dB] notated as w_1 were transferred to the untrained network in the medium-SNR region of [0 dB, 6 dB] as its initial weights. Similarly, the weights of the network trained with the signals in the medium-SNR region of [0 dB, 6 dB] notated as w_2 were transferred to the untrained network in the low-SNR region of [−8 dB, −2 dB] as its initial weights. The recognition accuracy comparison of the network with randomly initialized weights and the network with weight transfer is shown in Figure 14.

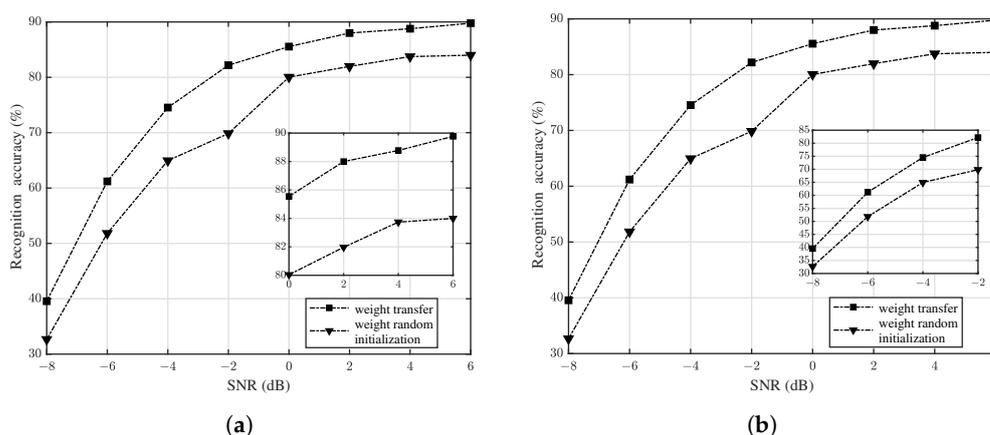


Figure 14. Recognition accuracy comparison between the network with randomly initialized weights and the network with weight transfer in the two lower-SNR regions: (a) [0 dB, 6 dB] and (b) [−8 dB, −2 dB].

In Figure 14, the circular line represents the accuracy curve of the network with randomly initialized weights, and the diamond line represents the accuracy curve of the network with the aforementioned w_1 as the initial weights. As can be seen in Figure 14a, the average recognition accuracy of the network with randomly initialized weights in the

medium-SNR region of [0 dB, 6 dB] was 82.44%, while the average accuracy of the network with weight transfer was 88.11%. The accuracy was improved by about 6%. Similarly, in Figure 14b, the average recognition accuracy of the network with randomly initialized weights in the low-SNR region of [−8 dB, −2 dB] was only 54.82%, while the network with weight transfer yielded better performance, with an average recognition rate of 65.63%, up by about 11%.

The confusion matrices of the 11 modulated signals in [0 dB, 6 dB] and [−8 dB, −2 dB] are displayed in Figure 15. It can be found that although the modulation recognition performance of the network with weight transfer in the two SNR regions was greatly improved, the four modulation types of AM-DSB, WBFM, 16QAM and 64QAM were still confused with each other.

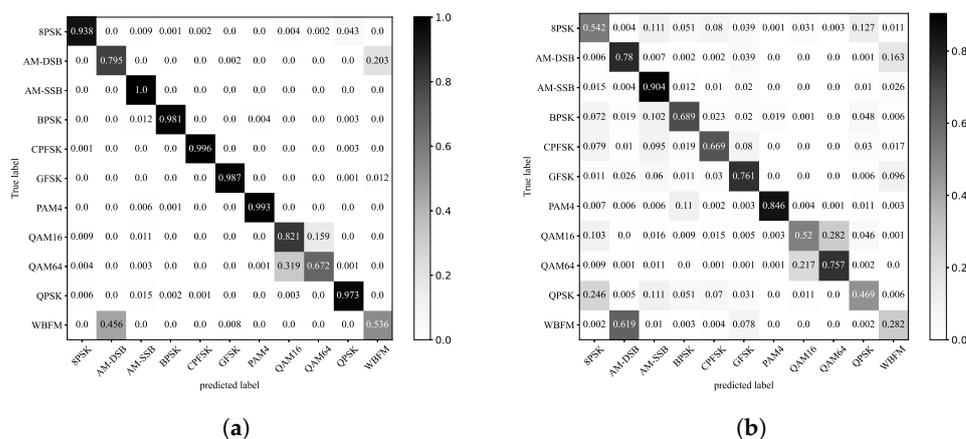


Figure 15. Confusion matrix of the network with weight transfer in the SNR regions of (a) [0 dB, 6 dB] and (b) [−8 dB, −2 dB].

5.4. Feature Re-Extraction

The deep features extracted by the backbone neural network could not effectively identify all modulated signals, and there were still some modulated signals that were confused. Therefore, based on the deep features extracted by the network, we considered re-extracting new, deeper features to obtain better recognition results. The above-mentioned 13 signal features, $F_0 \sim F_{12}$, were extracted, and the feature distribution curves of four confused modulated signals in three SNR regions are displayed in Figures 16–18.

As shown in the three figures, in the high-SNR region of [12 dB, 18 dB], only F_3, F_4, F_5 and F_{12} could distinguish among the four confused signals. In the medium-SNR region of [0 dB, 6 dB], F_3 could not distinguish among 16QAM, WBFM and 64QAM; F_5 could not distinguish between 16QAM and WBFM; and only F_4 and F_{12} could better distinguish among the four confused signals. In the low-SNR region of [−8 dB, −2 dB], it can be seen in the distribution curves of F_3, F_4 and F_5 that among the eigenvalue ranges of the four easily confused signals varying degrees of overlap existed. Only F_{12} could distinguish among the four confused signals.

In addition, the eigenvalue distributions of F_{12} in different SNR regions were more stable than the other three features, that is, all four eigenvalues of 16QAM and 64QAM were consistently larger than those of the other two signals, AM-DSB and WBFM. Therefore, the importance order of the four features was $F_{12} > F_4 > F_5 > F_3$, and the other nine features were not involved in the ranking because they could not distinguish among 16QAM, 64QAM, WBFM and AM-DSB even at high SNRs.

In Figure 19, for the signals in [12 dB, 18 dB], the modulation recognition accuracy of the network with randomly initialized weights was 91.96%, while the accuracy based on feature re-extraction was 98.66%, which is an improvement of about 7%. In [0 dB, 6 dB], the modulation recognition accuracy of the network with randomly initialized weights was 82.44%. The accuracy based on feature re-extraction was 95.95%, which is an

improvement of about 14%. In $[-8 \text{ dB}, -2 \text{ dB}]$, the modulation recognition accuracy of the network with randomly initialized weights was 54.82%, while the accuracy based on feature re-extraction was 71.68%, which is an improvement of about 17%. The experimental results reveal that the feature re-extraction method can greatly improve the modulation recognition performance.

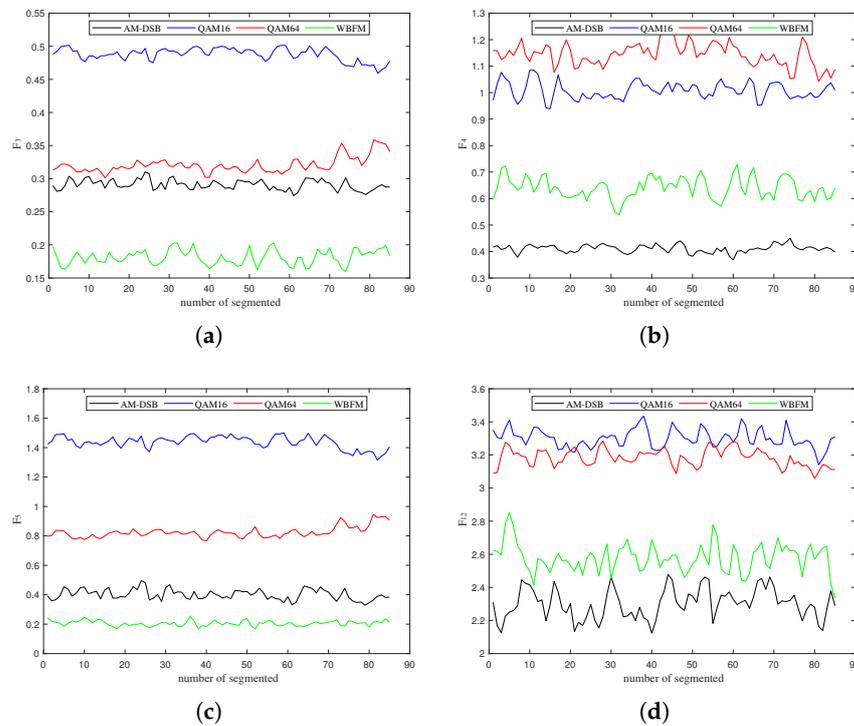


Figure 16. The distribution curves of features (a) F_3 , (b) F_4 , (c) F_5 and (d) F_{12} in $[12 \text{ dB}, 18 \text{ dB}]$.

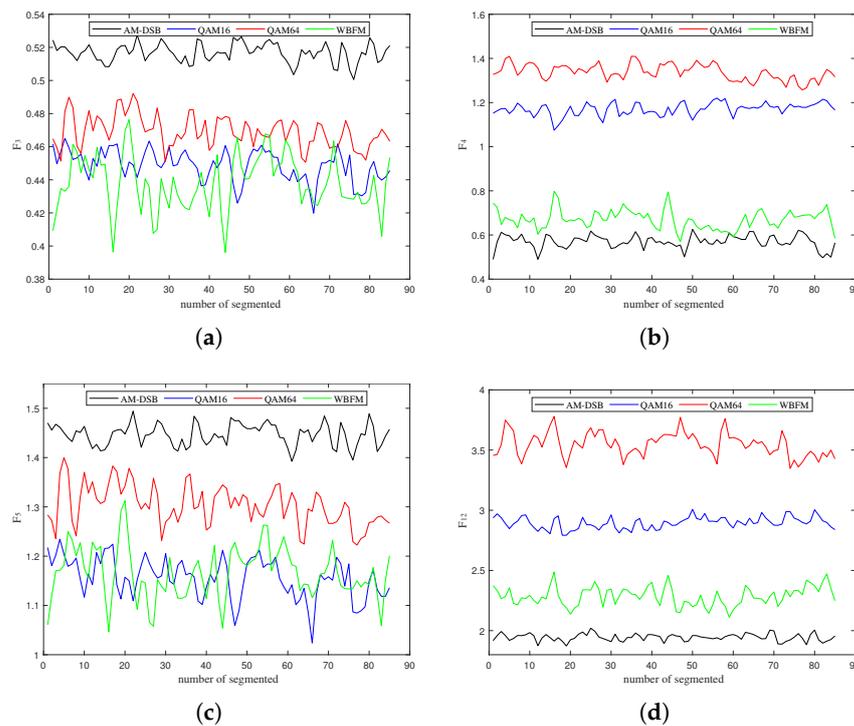


Figure 17. The distribution curves of features (a) F_3 , (b) F_4 , (c) F_5 and (d) F_{12} in $[0 \text{ dB}, 6 \text{ dB}]$.

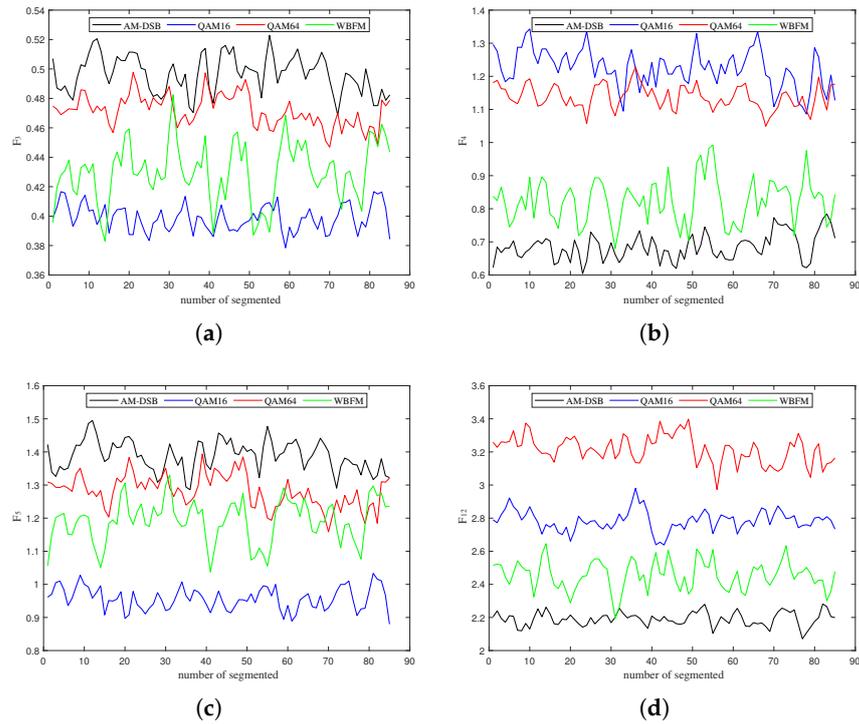


Figure 18. The distribution curves of features (a) F_3 , (b) F_4 , (c) F_5 and (d) F_{12} in $[-8 \text{ dB}, -2 \text{ dB}]$.

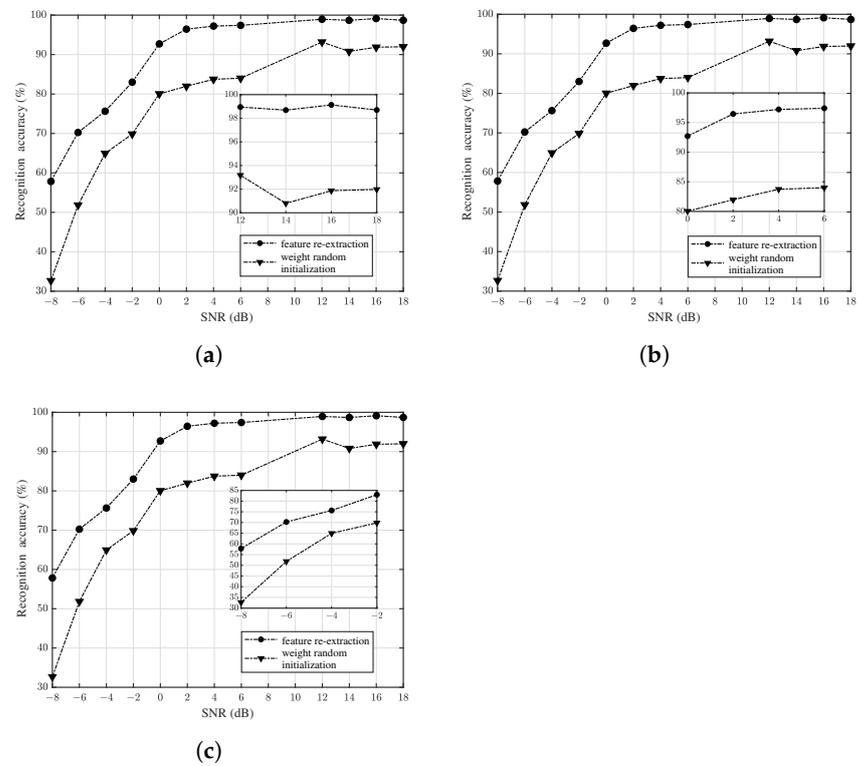


Figure 19. Recognition accuracy comparison before and after feature re-extraction by the network with randomly initialized weights in the three SNR regions: (a) $[12 \text{ dB}, 18 \text{ dB}]$, (b) $[0 \text{ dB}, 6 \text{ dB}]$ and (c) $[-8 \text{ dB}, -2 \text{ dB}]$.

Figure 20 shows the confusion matrices of the 11 modulated signals after feature re-extraction in the high-SNR region of $[12 \text{ dB}, 18 \text{ dB}]$. Seven modulation types could be

fully identified by the backbone neural network, and the other four confused modulation types, AM-DSB and WBFM, and 16QAM and 64QAM, could be well distinguished using feature re-extraction; thus, the effectiveness of the feature re-extraction method in confused signal recognition was further demonstrated.

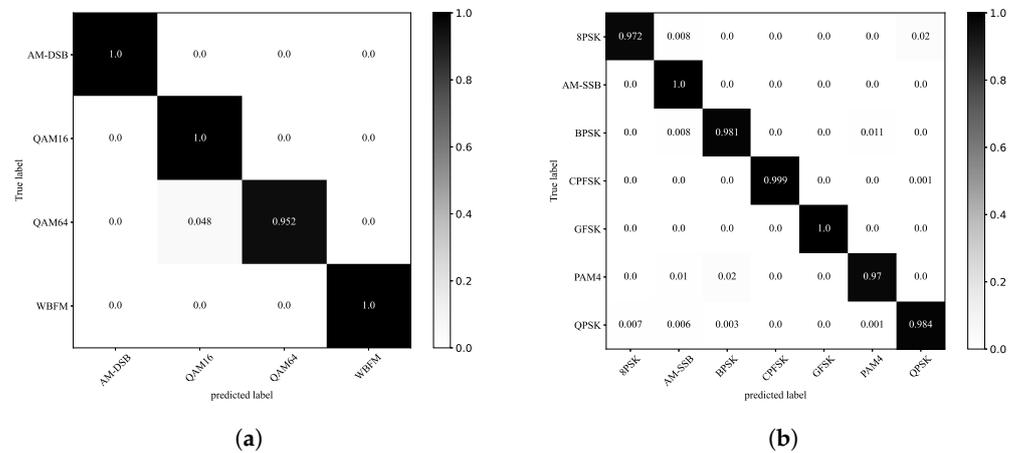


Figure 20. Confusion matrix of (a) the four confused modulation types and (b) the seven non-confused modulation types.

5.5. Combination of Fine-Tuning and Feature Re-Extraction

In this subsection, the comparison of modulation recognition results in the four cases of weight random initialization, weight transfer based on fine-tuning, feature re-extraction, and the combination of weight transfer and feature re-extraction in the two lower-SNR regions of [0 dB, 6 dB] and [−8 dB, −2 dB] is presented in Figure 21. In the figure, the triangular connection line represents the modulation recognition accuracy of the network with randomly initialized weights. The square connection line represents the modulation recognition accuracy of the network with weight transfer. The circular connection line represents the modulation recognition accuracy of feature re-extraction. The diamond connection line represents the modulation recognition accuracy of the combination of weight transfer and feature re-extraction. Figure 21a reveals that the average modulation recognition accuracy values of the above four cases were 82.44%, 88.11%, 95.95% and 98.23% in the medium-SNR region of [0 dB, 6 dB]. Similarly, Figure 21b reveals that the average modulation recognition accuracy values of the above four cases were 54.82%, 64.37%, 71.68% and 76.73% in the low-SNR region of [−8 dB, −2 dB].

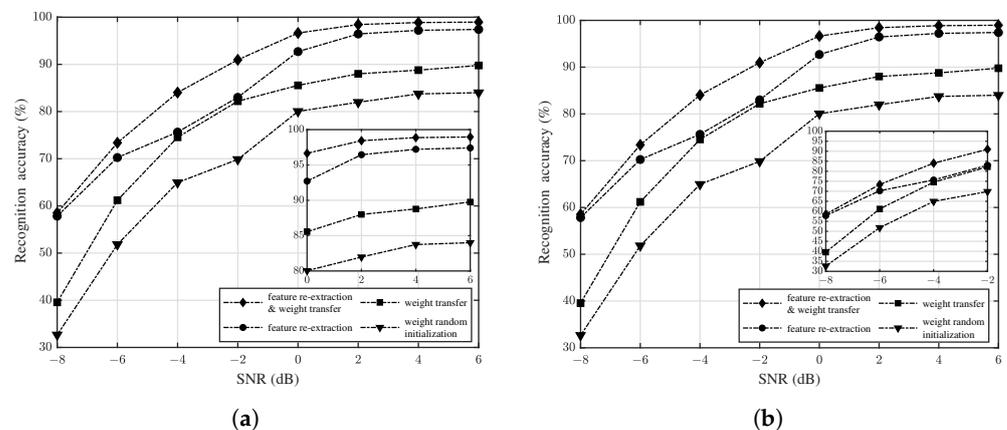


Figure 21. Recognition accuracy comparison in four cases in the two lower-SNR regions: (a) [0 dB, 6 dB] and (b) [−8 dB, −2 dB].

In summary, transfer learning and feature re-extraction can improve the modulation recognition performance to different extents, and feature re-extraction performs better. Moreover, combining fine-tuning and feature re-extraction can achieve the best modulation recognition accuracy.

5.6. Performance Comparison of the Proposed MR Algorithm and Other Existing Algorithms

In order to verify the superiority of the proposed method in terms of recognition accuracy and complexity, we compared the proposed algorithm with four existing algorithms, namely, CNN-IQ [59], LSTM-IQ [60], CLDNN-IQ [61] and MCLDNN-IQ [39]. IQ means that the network inputs are I-channel data and Q-channel data. We used the number of learned parameters and floating-point operations (FLOPs) as the measures of network complexity. The experimental results are presented in Figure 22 and Table 4.

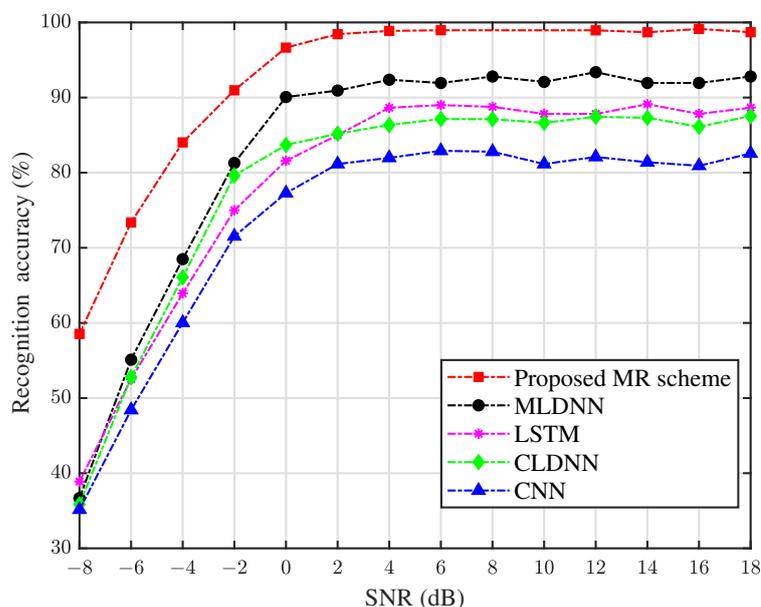


Figure 22. The comparison of the proposed algorithm and existing algorithms in terms of recognition accuracy.

Table 4. The comparison of the proposed algorithm and existing algorithms in terms of complexity.

MR Algorithm	Learned Parameters	FLOPs
CNN-IQ	129,867	130,132
LSTM-IQ	297,611	558,473
CLDNN-IQ	334,225	594,943
MCLDNN-IQ	405,175	665,742
Proposed algorithm	476,125	736,545

It can be observed in Figure 22 that CNN-IQ had relatively low classification accuracy. Its average recognition rate was 73.53% at SNRs ranging from -8 dB to 18 dB, and the maximum accuracy was 82.91% at SNR = 6 dB, which shows that CNN is relatively low-performing in the feature extraction of time-series signals. LSTM-IQ had better recognition results than CNN-IQ, but it has high computational requirements and requires a long training time to obtain good results. The CLDNN-IQ model had higher recognition accuracy than LSTM-IQ and CNN-IQ at low SNRs. The average accuracy of CLDNN-IQ was 78.49% at SNRs ranging from -8 dB to 18 dB, and the maximum accuracy was 87.44% at 12 dB. MCLDNN-IQ had better performance than CLDNN-IQ when $SNR \geq -6$ dB. Its maximum recognition accuracy was 93.38% at SNR = 12 dB, and the average accuracy was 82.99% at SNRs ranging from -8 dB to 18 dB. Regarding the optimization scheme proposed

in this paper, its recognition accuracy reached 58.53% at SNR = −8 dB and 96.65% at SNR = 0 dB. The average accuracy was 91.28% at SNRs ranging from −8 dB to 18 dB, which is an improvement of 8% to 17% compared with that of the other four existing schemes. The simulation results prove that the proposed scheme is an advanced method in terms of recognition accuracy.

Table 4 provides the comparison of the proposed algorithm and existing algorithms in terms of complexity. It can be seen that the network complexity of the five algorithms, including CNN-IQ, LSTM-IQ, CLDNN-IQ, MCLDNN-IQ and the proposed algorithm, sequentially increases. Although the complexity of the proposed algorithm is the highest, the increase is slight compared with MCLDNN-IQ. Moreover, considering the significant improvement in recognition accuracy, the complexity of the proposed algorithm is within the acceptable limits.

6. Conclusions

In this paper, an optimization scheme for deep learning-based modulation recognition algorithms is proposed to address the fact that neural networks have poor recognition effects in some cases. Using transfer learning, the knowledge (network weights) learned from high-SNR data can be transferred to the low-SNR network, which improves the scalability and robustness of the network at low SNRs. The deeper and more discriminative feature representation of the original signal can be obtained using feature re-extraction, thereby effectively identifying confused modulation types. In addition, the improved recognition method combining deep learning and traditional machine learning or the likelihood ratio test will be investigated in future work.

Author Contributions: Conceptualization, L.Z., Z.Y. (Zhutian Yang) and L.W.; Methodology, Y.W., Z.Y. (Zhendong Yin) and Y.Z.; Software, Y.W.; Validation, Y.W.; Writing—original draft preparation, Y.W., L.Z. and Z.Y. (Zhutian Yang); Writing—review and editing, Z.Y. (Zhutian Yang), Z.Y. (Zhendong Yin) and Y.Z.; Project administration, L.W. and Y.Z.; Funding acquisition, Z.W. and Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by General Program of National Natural Science Foundation of China (grant No. 62071143) and by National Natural Science Foundation of China (grant No. 62071153).

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the editor-in-chief, the editor, and the anonymous reviewers for their valuable reviews.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dobre, O.A.; Abdi, A.; Bar-Ness, Y.; Su, W. Survey of automatic modulation classification techniques: Classical approaches and new trends. *IET Commun.* **2007**, *1*, 137–156. [[CrossRef](#)]
2. Huang, Z.; Yang, J.; Wang, X.; Cui, X.; Wang, Y. A Survey of modulation recognition algorithms in non-cooperative communication. *Sci. Technol. Rev.* **2019**, *37*, 55–62. [[CrossRef](#)]
3. Zebarjadi, M.; Teimouri, M. Non-cooperative burst detection and synchronisation in downlink TDMA-based wireless communication networks. *IET Commun.* **2019**, *13*, 863–872. [[CrossRef](#)]
4. Liu, C.; Wang, H.; Zhang, J.; He, Z. Wideband spectrum sensing based on single-channel sub-Nyquist sampling for cognitive radio. *Sensors* **2018**, *18*, 2222. [[CrossRef](#)]
5. Yang, Z.; Li, D.; Zhao, N.; Wu, Z.; Li, Y.; Niyato, D. Secure precoding optimization for NOMA-aided integrated sensing and communication. *IEEE Trans. Commun.* **2022**, *70*, 8370–8382. [[CrossRef](#)]
6. Ma, R.; Tang, J.; Zhang, X.; Wong, K.K.; Chambers, J. Energy Efficiency Optimization for Mutual-Coupling-Aware Wireless Communication System based on RIS-enhanced SWIPT. *IEEE Internet Things J.* **2023**. [[CrossRef](#)]
7. Liu, M.; Zhang, H.; Liu, Z.; Zhao, N. Attacking Spectrum Sensing With Adversarial Deep Learning in Cognitive Radio-Enabled Internet of Things. *IEEE Trans. Reliab.* **2022**, 1–14. [[CrossRef](#)]
8. Mohammed Tag Elsir Awad, E.; Xiong, Y.; Wang, J.; Tang, B. A new approach for high order MQAM signal modulation recognition. In Proceedings of the Eighth International Conference on Digital Image Processing (ICDIP 2016), Chengdu, China, 20–23 May 2016; pp. 1036–1040. [[CrossRef](#)]

9. Kumar, A.; Majhi, S.; Gui, G.; Wu, H.C.; Yuen, C. A Survey of Blind Modulation Classification Techniques for OFDM Signals. *Sensors* **2022**, *22*, 1020. [[CrossRef](#)]
10. Kim, K.; Polydoros, A. Digital modulation classification: The BPSK versus QPSK case. In Proceedings of the MILCOM 88, 21st Century Military Communications-What's Possible?. Conference Record. Military Communications Conference, San Diego, CA, USA, 23–26 October 1988; pp. 431–436. [[CrossRef](#)]
11. Wei, W.; Mendel, J.M. Maximum-likelihood classification for digital amplitude-phase modulations. *IEEE Trans. Commun.* **2000**, *48*, 189–193. [[CrossRef](#)]
12. Majhi, S.; Gupta, R.; Xiang, W.; Glisic, S. Hierarchical hypothesis and feature-based blind modulation classification for linearly modulated signals. *IEEE Trans. Veh. Technol.* **2017**, *66*, 11057–11069. [[CrossRef](#)]
13. Gupta, R.; Majhi, S.; Dobre, O.A. Design and implementation of a tree-based blind modulation classification algorithm for multiple-antenna systems. *IEEE Trans. Instrum. Meas.* **2018**, *68*, 3020–3031. [[CrossRef](#)]
14. Zhang, X.; Ge, T.; Chen, Z. Automatic modulation recognition of communication signals based on instantaneous statistical characteristics and SVM classifier. In Proceedings of the 2018 IEEE Asia-Pacific Conference on Antennas and Propagation (APCAP), Auckland, New Zealand, 5–8 August 2018; pp. 344–346. [[CrossRef](#)]
15. Yang, Y.; Yang, L.; Hu, M. A method for digital modulation recognition based on mixed signal features. In Proceedings of the 2019 International Conference on Electronic Engineering and Informatics (EEI), Nanjing, China, 8–10 November 2019; pp. 195–199. [[CrossRef](#)]
16. Xie, W.; Hu, S.; Yu, C.; Zhu, P.; Peng, X.; Ouyang, J. Deep learning in digital modulation recognition using high order cumulants. *IEEE Access* **2019**, *7*, 63760–63766. [[CrossRef](#)]
17. Wang, A.; Li, R. Research on digital signal recognition based on higher order cumulants. In Proceedings of the 2019 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), Changsha, China, 12–13 January 2019; pp. 586–588. [[CrossRef](#)]
18. Hassanpour, S.; Pezeshk, A.M.; Behnia, F. A robust algorithm based on wavelet transform for recognition of binary digital modulations. In Proceedings of the 2015 38th International Conference on Telecommunications and Signal Processing (TSP), Prague, Czech Republic, 9–11 July 2015; pp. 508–512. [[CrossRef](#)]
19. Yang, J.; Liu, F. Modulation recognition using wavelet transform based on AlexNet. In Proceedings of the 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT), Dalian, China, 19–20 October 2019; pp. 339–342. [[CrossRef](#)]
20. Majhi, S.; Gupta, R.; Xiang, W. Novel blind modulation classification of circular and linearly modulated signals using cyclic cumulants. In Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, Canada, 8–13 October 2017; pp. 1–5. [[CrossRef](#)]
21. Wei, Y.; Fang, S.; Wang, X. Automatic modulation classification of digital communication signals using SVM based on hybrid features, cyclostationary, and information entropy. *Entropy* **2019**, *21*, 745. [[CrossRef](#)] [[PubMed](#)]
22. Shi, C. Signal pattern recognition based on fractal features and machine learning. *Appl. Sci.* **2018**, *8*, 1327. [[CrossRef](#)]
23. Wang, F.; Wang, Y.; Chen, X. Graphic constellations and DBN based automatic modulation classification. In Proceedings of the 2017 IEEE 85th Vehicular Technology Conference (VTC Spring), Sydney, Australia, 4–7 June 2017; pp. 1–5. [[CrossRef](#)]
24. Yan, X.; Zhang, G.; Wu, H.C. A Novel Automatic Modulation Classifier Using Graph-Based Constellation Analysis for M-ary QAM. *IEEE Commun. Lett.* **2018**, *23*, 298–301. [[CrossRef](#)]
25. Liu, M.; Liu, C.; Chen, Y.; Yan, Z.; Zhao, N. Radio Frequency Fingerprint Collaborative Intelligent Blind Identification for Green Radios. *IEEE Trans. Green Commun. Netw.* **2022**. [[CrossRef](#)]
26. Ding, Y.; Feng, Y.; Lu, W.; Zheng, S.; Zhao, N.; Meng, L.; Nallanathan, A.; Yang, X. Online Edge Learning Offloading and Resource Management for UAV-Assisted MEC Secure Communications. *IEEE J. Sel. Top. Signal Process.* **2023**, *17*, 54–65. [[CrossRef](#)]
27. Lu, W.; Mo, Y.; Feng, Y.; Gao, Y.; Zhao, N.; Wu, Y.; Nallanathan, A. Secure Transmission for Multi-UAV-Assisted Mobile Edge Computing Based on Reinforcement Learning. *IEEE Trans. Netw. Sci. Eng.* **2022**, 1–12. [[CrossRef](#)]
28. Peng, S.; Jiang, H.; Wang, H.; Alwageed, H.; Yao, Y. Modulation classification using convolutional neural network based deep learning model. In Proceedings of the 2017 26th Wireless and Optical Communication Conference (WOCC), Newark, NJ, USA, 7–8 April 2017; pp. 1–5. [[CrossRef](#)]
29. Matuszewski, J.; Pietrow, D. Recognition of electromagnetic sources with the use of deep neural networks. In Proceedings of the XII Conference on Reconnaissance and Electronic Warfare Systems, Oltarzew, Poland, 19–21 November 2018; Volume 11055, pp. 100–114. [[CrossRef](#)]
30. Matuszewski, J.; Pietrow, D. Specific radar recognition based on characteristics of emitted radio waveforms using convolutional neural networks. *Sensors* **2021**, *21*, 8237. [[CrossRef](#)]
31. Lee, J.; Kim, B.; Kim, J.; Yoon, D.; Choi, J.W. Deep neural network-based blind modulation classification for fading channels. In Proceedings of the 2017 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, 18–20 October 2017; pp. 551–554. [[CrossRef](#)]
32. Kim, B.; Kim, J.; Chae, H.; Yoon, D.; Choi, J.W. Deep neural network-based automatic modulation classification technique. In Proceedings of the 2016 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Republic of Korea, 19–21 October 2016; pp. 579–582. [[CrossRef](#)]

33. Mendis, G.J.; Wei, J.; Madanayake, A. Deep learning-based automated modulation classification for cognitive radio. In Proceedings of the 2016 IEEE International Conference on Communication Systems (ICCS), Shenzhen, China, 14–16 December 2016; pp. 1–6. [[CrossRef](#)]
34. Duan, S.; Chen, K.; Yu, X.; Qian, M. Automatic multicarrier waveform classification via PCA and convolutional neural networks. *IEEE Access* **2018**, *6*, 51365–51373. [[CrossRef](#)]
35. Gou, Z.; Xu, H.; Zheng, W.; Feng, L.; Bai, P. Semi-supervised Joint Neural Network Based Recognition Algorithm of Modulation Signal. *J. Signal Process.* **2020**, *36*, 168–176. [[CrossRef](#)]
36. O’Shea, T.J.; Roy, T.; Clancy, T.C. Over-the-air deep learning based radio signal classification. *IEEE J. Sel. Top. Signal Process.* **2018**, *12*, 168–179. [[CrossRef](#)]
37. Zhang, Y.; Liu, T.; Zhang, L.; Wang, K. A deep learning approach for modulation recognition. In Proceedings of the 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), Shanghai, China, 19–21 November 2018; pp. 1–5. [[CrossRef](#)]
38. Vanhoy, G.; Thurston, N.; Burger, A.; Breckenridge, J.; Bose, T. Hierarchical modulation classification using deep learning. In Proceedings of the MILCOM 2018—2018 IEEE Military Communications Conference (MILCOM), Los Angeles, CA, USA, 29–31 October 2018; pp. 20–25. [[CrossRef](#)]
39. Xu, J.; Luo, C.; Parr, G.; Luo, Y. A spatiotemporal multi-channel learning framework for automatic modulation recognition. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 1629–1632. [[CrossRef](#)]
40. Bu, K.; He, Y.; Jing, X.; Han, J. Adversarial transfer learning for deep learning based automatic modulation classification. *IEEE Signal Process. Lett.* **2020**, *27*, 880–884. [[CrossRef](#)]
41. Wang, Y.; Gui, G.; Gacanin, H.; Ohtsuki, T.; Sari, H.; Adachi, F. Transfer learning for semi-supervised automatic modulation classification in ZF-MIMO systems. *IEEE J. Emerg. Sel. Top. Power Electron.* **2020**, *10*, 231–239. [[CrossRef](#)]
42. Perenda, E.; Rajendran, S.; Bovet, G.; Pollin, S.; Zheleva, M. Learning the unknown: Improving modulation classification performance in unseen scenarios. In Proceedings of the IEEE INFOCOM 2021—IEEE Conference on Computer Communications, Vancouver, BC, Canada, 10–13 May 2021; pp. 1–10. [[CrossRef](#)]
43. Pauluzzi, D.R.; Beaulieu, N.C. A comparison of SNR estimation techniques for the AWGN channel. *IEEE Trans. Commun.* **2000**, *48*, 1681–1691. [[CrossRef](#)]
44. Qun, X.; Jian, Z. Improved SNR estimation algorithm. In Proceedings of the 2017 International Conference on Computer Systems, Electronics and Control (ICCSEC), Dalian, China, 25–27 December 2017; pp. 1458–1461. [[CrossRef](#)]
45. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [[CrossRef](#)]
46. Niu, S.; Liu, Y.; Wang, J.; Song, H. A decade survey of transfer learning (2010–2020). *IEEE Trans. Artif. Intell.* **2020**, *1*, 151–166. [[CrossRef](#)]
47. Vrbančič, G.; Podgorelec, V. Transfer learning with adaptive fine-tuning. *IEEE Access* **2020**, *8*, 196197–196211. [[CrossRef](#)]
48. O’Shea, T.J.; West, N. Radio machine learning dataset generation with GNU radio. In Proceedings of the GNU Radio Conference 2016, Boulder, CO, USA, 12–16 September 2017; pp. 1–6.
49. Li, J.; Cheng, J.H.; Shi, J.Y.; Huang, F. Brief introduction of back propagation (BP) neural network algorithm and its improvement. In Proceedings of the Advances in Computer Science and Information Engineering, Zhengzhou, China, 19–20 May 2012; Volume 2, pp. 553–558. [[CrossRef](#)]
50. Zaheer, R.; Shaziya, H. A study of the optimization algorithms in deep learning. In Proceedings of the 2019 Third International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 10–11 January 2019; pp. 536–539. [[CrossRef](#)]
51. Abdelmutalab, A.; Assaleh, K.; El-Tarhuni, M. Automatic modulation classification based on high order cumulants and hierarchical polynomial classifiers. *Phys. Commun.* **2016**, *21*, 10–18. [[CrossRef](#)]
52. Zhao, Y.; Xu, Y.T.; Jiang, H.; Luo, Y.J.; Wang, Z.W. Recognition of digital modulation signals based on high-order cumulants. In Proceedings of the 2015 International Conference on Wireless Communications & Signal Processing (WCSP), Nanjing, China, 15–17 October 2015; pp. 1–5. [[CrossRef](#)]
53. Hao, Y.; Wang, X.; Lan, X. Frequency domain analysis and convolutional neural network based modulation signal classification method in OFDM system. In Proceedings of the 2021 13th International Conference on Wireless Communications and Signal Processing (WCSP), Changsha, China, 20–22 October 2021; pp. 1–5. [[CrossRef](#)]
54. Bai, J.; Gao, L.; Gao, J.; Li, H.; Zhang, R.; Lu, Y. A new radar signal modulation recognition algorithm based on time-frequency transform. In Proceedings of the 2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP), Wuxi, China, 19–21 July 2019; pp. 21–25. [[CrossRef](#)]
55. Venkata Subbarao, M.; Samundiswary, P. Spectrum sensing in cognitive radio networks using time–frequency analysis and modulation recognition. In Proceedings of the Microelectronics, Electromagnetics and Telecommunications: Proceedings of ICMEET 2017, Hyderabad, India, 9–10 September 2017; pp. 827–837. [[CrossRef](#)]
56. Zeng, Y.; Zhang, M.; Han, F.; Gong, Y.; Zhang, J. Spectrum analysis and convolutional neural network for automatic modulation recognition. *IEEE Wirel. Commun. Lett.* **2019**, *8*, 929–932. [[CrossRef](#)]
57. Zhang, L.; Liu, H.; Yang, X.; Jiang, Y.; Wu, Z. Intelligent denoising-aided deep learning modulation recognition with cyclic spectrum features for higher accuracy. *IEEE Trans. Aerosp. Electron. Syst.* **2021**, *57*, 3749–3757. [[CrossRef](#)]
58. Nandi, A.K.; Azzouz, E.E. Algorithms for automatic modulation recognition of communication signals. *IEEE Trans. Commun.* **1998**, *46*, 431–436. [[CrossRef](#)]

59. O'Shea, T.J.; Corgan, J.; Clancy, T.C. Convolutional radio modulation recognition networks. In Proceedings of the Engineering Applications of Neural Networks, Aberdeen, UK, 2–5 September 2016; pp. 213–226. [\[CrossRef\]](#)
60. Rajendran, S.; Meert, W.; Giustiniano, D.; Lenders, V.; Pollin, S. Deep learning models for wireless signal classification with distributed low-cost spectrum sensors. *IEEE Trans. Cogn. Commun. Netw.* **2018**, *4*, 433–445. [\[CrossRef\]](#)
61. West, N.E.; O'shea, T. Deep architectures for modulation recognition. In Proceedings of the 2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), Baltimore, MD, USA, 6–9 March 2017; pp. 1–6. [\[CrossRef\]](#)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.