drones

Article

Decentralized Learning and Model Averaging Based Automatic Modulation Classification in Drone Communication Systems

Min Ma [1], Yunhe Xu [2], Zhi Wang [1], Xue Fu [2] and Guan Gui [2,*]

[1] School of Network and Communication, Nanjing Vocational College of Information Technology, Nanjing 210023, China; mamin@njcit.cn (M.M.); wangzhi@njcit.cn (Z.W.)
[2] College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; 1320017410@njupt.edu.cn (Y.X.)
* Correspondence: guiguan@njupt.edu.cn

Abstract: Automatic modulation classification (AMC) is a promising technology to identify the modulation mode of the received signal in drone communication systems. Recently, benefiting from the outstanding classification performance of deep learning (DL), various deep neural networks (DNNs) have been introduced into AMC methods. Most current AMC methods are based on a local framework (LocalAMC) where there is only one device, or a centralized framework (CentAMC) where multiple local devices (LDs) upload their data to only one central server (CS). LocalAMC may not achieve ideal results due to insufficient data and finite computational power. CentAMC carries a significant risk of privacy leakage and the final data for training model in CS are quite massive. In this paper, we propose a practical and light AMC method based on decentralized learning with residual network (ResNet) in drone communication systems. Simulation results show that the ResNet-based decentralized AMC (DecentAMC) method achieves similar classification performance to CentAMC while improving training efficiency and protecting data privacy.

Keywords: drone communications; automatic modulation classification (AMC); deep learning; deep neural network; decentralized learning; residual network (ResNet)

check for updates

Citation: Ma, M.; Xu, Y.; Wang, Z.; Fu, X.; Gui, G. Decentralized Learning and Model Averaging Based Automatic Modulation Classification in Drone Communication Systems. *Drones* **2023**, *7*, 391. https://doi.org/10.3390/drones7060391

Academic Editor: Petros S. Bithas

Received: 1 May 2023
Revised: 5 June 2023
Accepted: 9 June 2023
Published: 12 June 2023

1. Introduction

Advanced signal processing techniques accelerate the fast development of future wireless communications. Among these techniques, automatic modulation classification (AMC) is considered a promising technique for identifying signals in complicated electromagnetic environments such as in drone communication systems [1–4]. With the ability to identify modulation mode of the received or intercepted signal without prior modulation information [5], AMC has been widely considered in various scenarios. Originally, AMC has been located in military fields, such as drone recognition, signal monitoring, interference recognition, electronic countermeasures [6–8] and so on. With the constant consumption of spectrum resources, AMC has been introduced into civilian scenarios. In these scenarios, the signal transmitters do not need to send additional modulation information of signals, which means saving spectrum resources [9].

Classic AMC methods consist of likelihood-based (LB) methods and feature-based (FB) methods. The main LB methods can be classified as average likelihood ratio rest (ALRT) [10], generalized likelihood ratio test (GLRT) [11] and hybrid likelihood ratio test (HLRT) [12]. Other later LB methods are variations of above three AMC methods. According to the Bayes minimum misclassification cost criterion, LB methods can theoretically guarantee good classification performance. However, LB methods cannot achieve ideal results in non-cooperative communication due to the lack of prior knowledge. FB methods consist of extracting features and constructing classifiers [13–15]. Classic man-made features consist of parametric statistical features [16], high order statistical features, constellation features [17], wavelet transform features [18] and cyclic putter features [19,20]. In FB

AMC, typical machine learning (ML)-based AMC methods were proposed first. These methods are simple in principle, easy to implement and can achieve excellent classification performance, but feature extraction requires too much artificial cost.

Compared with ML, deep learning (DL) has the advantage of automatically extracting depth features from the training samples and making use of these depth features to build a classifier for decision making. Consequently, DL is seen as one of the most promising algorithms in classification tasks. M. Liu et al. [21] proposed a message passing algorithm based on DL for efficient resource allocation in cognitive radio networks. J. Tan et al. [22] proposed a deep reinforcement learning approach for sharing intelligently unlicensed bands in long term evolution (LTE) and wireless fidelity (WiFi) systems. H. Huang et al. [23] proposed a fast unsupervised DL-based beamforming technology. There are also proposed AMC methods combined with DL. It was O'shea who introduced convolutional neural network (CNN) into AMC firstly [24] and proved the superiority of DL in classification tasks. Y. Lin et al. [25] proposed a neural network pruning technology for AMC. S. Huang et al. [26] proposed an AMC method utilizing a compressive CNN structure. Z. Zhang et al. [27] proposed an AMC method using CNN with feature fusion. Y. Wang et al. [28] proposed an AMC method with compressive sensing, which introduced a scaling factor for each neuron in CNN. F. Meng et al. [29] proposed a DL-enabled approach for AMC, whose computation speed is faster than traditional likelihood-based AMC methods. Z. Cao et al. [30] proposed a light-weight CNN for channel statement information (CSI) feedback in massive MIMO. P. Qi et al. [31] proposed an AMC method based on deep residual network with multimodal information. The above AMC methods are all under the scope of LocalAMC or CentAMC. LocalAMC may not achieve ideal classification performance because the dataset may be not sufficient and the local device (LD) has limited computational power. The training efficiency of CentAMC is low because only one central server (CS) is used to process abundant data which are collected from local devices (LDs). Transmission of training data consumes a lot of communication overhead. In addition, CentAMC has the problem of privacy leakage during data transmission.

From the perspective of training efficiency and data security, decentralized or distributed learning algorithms [32,33], are applied in AMC research. Decentralized AMC (DecentAMC) enables learning a global model while datasets are distributed across LDs instead of gathered into only one CS, and LD only needs to transfer the local model to the CS. Wang et al. [34] proposed a DecentAMC method based on the CNN structure and proved that the training efficiency of DecentAMC is higher than that of CentAMC. Fu et al. [35] proposed a DecentAMC method using lightweight network and model aggregation which can be considered in the application of drone communications. Most current DecentAMC methods are based on the CNN structure. However, with the deepening of convolution layers, there will be gradient explosion and gradient disappearance problems. In order to solve the above problems, He et al. [36] proposed residual network (ResNet). Considering the advantages of decentralized learning and ResNet, we propose a decentralized learning method for AMC in the drons communication systems based on ResNet, which is termed ResNet-based DecentAMC. Our main contributions of this paper are highlighted below.

- We propose an AMC method using decentralized learning and residual network (ResNet) towards drone communication systems. This novel framework can achieve good classification performance and improve training efficiency while protecting data privacy.
- We compare the classification accuracy of support vector machine (SVM)-based CentAMC method and deep neural network (DNN)-based CentAMC method using dataset RadioML 2016.10a, and improve that DL-based AMC performs better than ML-based AMC.
- We compare classification accuracy of different DNN-based AMC methods using dataset RadioML 2018.01a. The proposed ResNet-based DecentAMC method performs better than current DNN-based DecentAMC method.

The remainder of this paper is structured as follows. We introduce the system framework of AMC and signal model in Section 2. The traditional AMC method based on artificial features is introduced in Section 3 to compare with DL-based AMC methods. The proposed DL-based AMC framework is introduced in Section 4. Section 5 presents and analyses the simulation results, and the conclusion is presented in Section 6.

2. AMC Description and Signal Model

2.1. AMC Description

AMC is used to classify the modulated mode of intercepted or received signals in military scenarios or civilian scenarios such as in drone communication systems. Specifically, the transmitter sends modulated signals. Through the wireless channel, the modulated signals are received and pre-processed for modulation recognition in the receiver. Then, the modulation recognition results are used to assist the demodulation of signals. The system framework of AMC is shown in Figure 1.

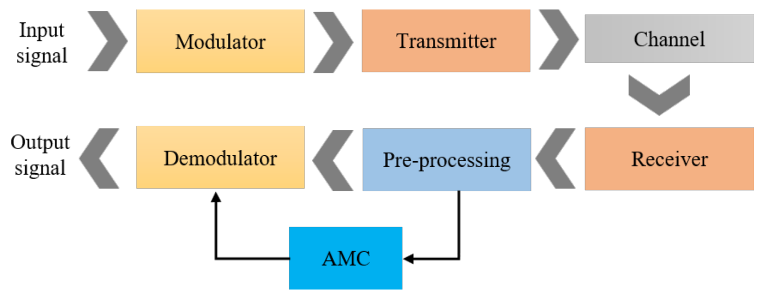


Figure 1. A basic system framework of AMC.

2.2. Signal Model

The band signal considered in this paper can be expressed as

$$r(k) = he^{j(2\pi f_0 k+\phi_0)}s_m(k) + w(k), k = 0, 1, \cdots, K-1. \tag{1}$$

Please notice that the specific definition of the model is shown in Table 1.

Table 1. Specific definition of the signal model.

Parameter	Definition
$r(k)$	The received band signal
h	Channel coefficient
f_0	Frequency offset
ϕ_0	Carrier phase offset
$s_m(k)$	k-th symbol generated
m of $s_m(k)$	m-th modulation scheme
$w(k)$	Additive Gaussian noise
K	The number of signal symbols

3. ML-Based AMC and DL-Based AMC

3.1. Classic AMC Method Based on Artificial Features and ML

The classic FB methods include pre-processing, extracting features, constructing the classifier and so on. Here, we consider high order statistical features, which can be expressed as [37]

$$\hat{C}_{20} = M_{20}, \tag{2}$$

$$\hat{C}_{21} = M_{21}, \tag{3}$$

$$\hat{C}_{40} = M_{40} - 3{M_{20}}^2, \tag{4}$$

$$\hat{C}_{42} = M_{42} - |M_{20}|^2 - 2M_{21}^2, \tag{5}$$

$$\hat{C}_{60} = M_{60} - 15|M_{40}||M_{20}| + 30M_{20}^3, \tag{6}$$

$$\hat{C}_{63} = M_{63} - 6M_{41}M_{20} - 9M_{42}M_{21} + 18M_{20}^2 M_{21} + 12M_{21}^3, \tag{7}$$

where number of the sampling points of the received signal $r = \{r(k)\}_{k=1}^{K}$ is limited in the real-world scenarios, thus we apply the estimation value of i-th order moments $\hat{M}_{ij} = 1/K \sum_{k=1}^{K} [r^{i-j}(k) r^{*j}(k)]$ to replace its theoretical value M_{ij}. As shown in Table 2, $|\hat{C}_{21}|$ of 8PSK, BPSK, QAM16, QAM64, and QPSK are equal. If $|\hat{C}_{40}|$, $|\hat{C}_{42}|$ and $|\hat{C}_{63}|$ are used as features, 8PSK, BPSK, QAM16, QAM64 and QPSK can be theoretically classified. Therefore, we use $F_1 = \frac{|\hat{C}_{40}|}{|\hat{C}_{21}|^2}$, $F_2 = \frac{|\hat{C}_{42}|}{|\hat{C}_{21}|^2}$ and $F_3 \frac{|\hat{C}_{63}|}{|\hat{C}_{21}|^3}$ as features and SVM as the classifier for machine-learning-based AMC.

Table 2. Theoretical Values of High-order Cumulant for Each Modulation Signal.

	$\lvert\hat{C}_{20}\rvert$	$\lvert\hat{C}_{21}\rvert$	$\lvert\hat{C}_{40}\rvert$	$\lvert\hat{C}_{41}\rvert$	$\lvert\hat{C}_{42}\rvert$	$\lvert\hat{C}_{60}\rvert$	$\lvert\hat{C}_{63}\rvert$
BPSK	E	E	$2E^2$	$2E^2$	$2E^2$	$16E^2$	$16E^3$
QPSK	0	E	E^2	0	E^2	0	$4E^3$
8PSK	0	E	0	0	E^2	0	$4E^3$
16QAM	0	E	$0.68E^2$	0	$0.68E^2$	0	$2.08E^3$
64QAM	0	E	$0.62E^2$	0	$0.62E^2$	0	$1.80E^3$

3.2. Modern AMC Method Based on Deep Features and DL

DL has the advantage of automatically extracting features from samples. In order to verify the universality that DL-based AMC methods perform better than ML-based AMC methods in classification performance, we consider comparing the classification performance of SVM and three DNNs which include CNN, modulation classification network (MCNet) [38] and ResNet. The structures of the above three DNNs are shown in Figures 2–4, respectively.

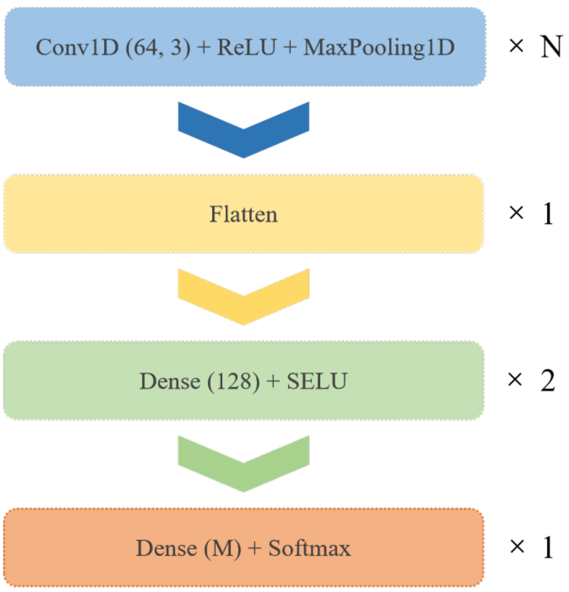


Figure 2. Structure of CNN, where the values of N and M are 1 and 5 for RadioML 2016.10a respectively, and 6 and 24 for RadioML 2018.01a, respectively.

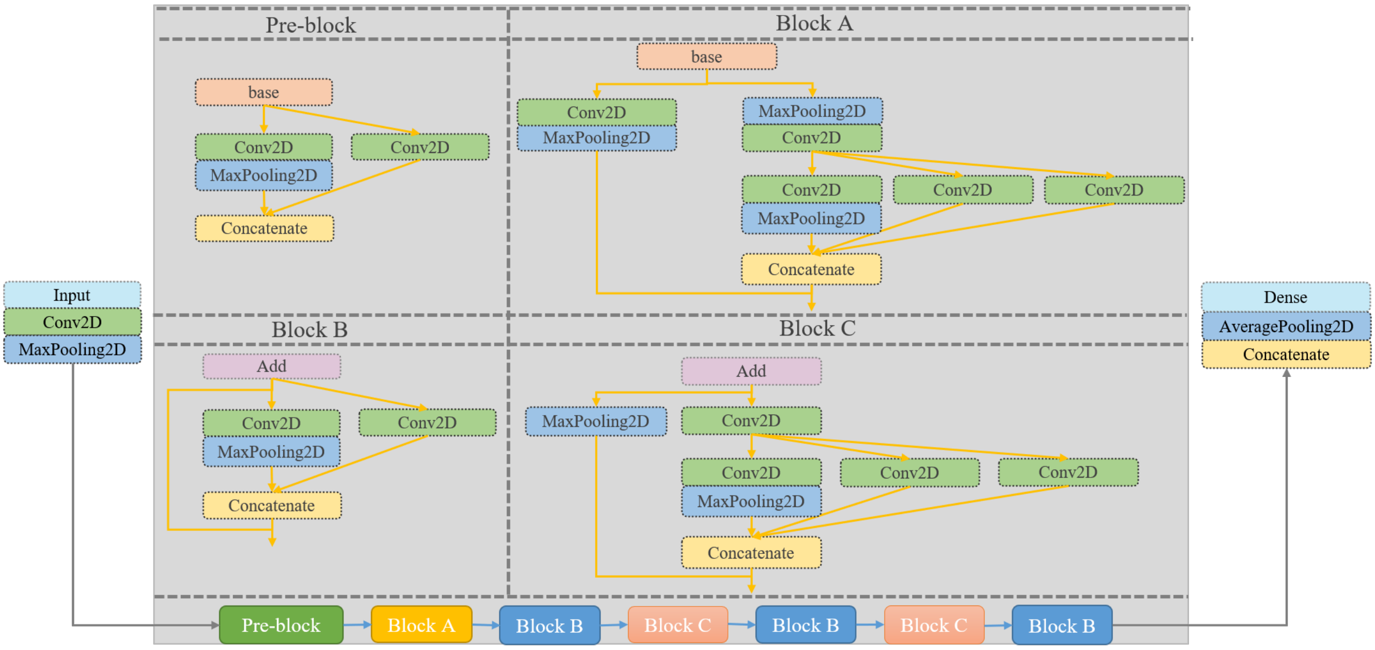


Figure 3. Structure of MCNet.

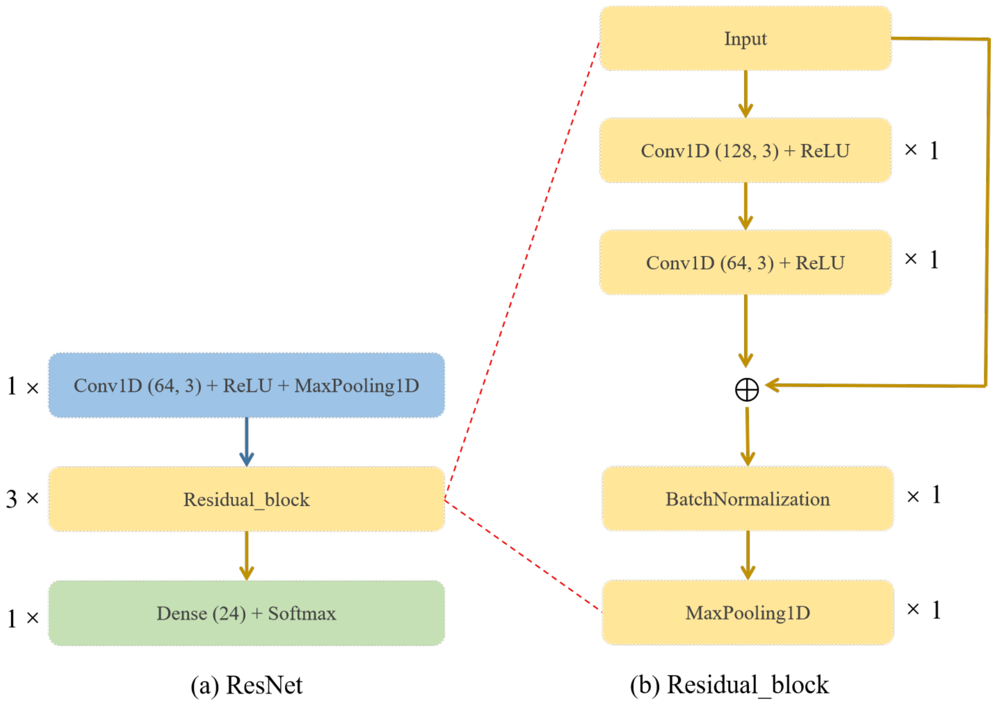


Figure 4. Structure of ResNet.

4. Our Proposed AMC Method

4.1. System Model of DecentAMC

DecentAMC consists of four steps: broadcasting the initial comprehensive model; training, updating and uploading the local model; local model aggregation and global model downloading. The specific system model of DecentAMC is shown in Figure 5.

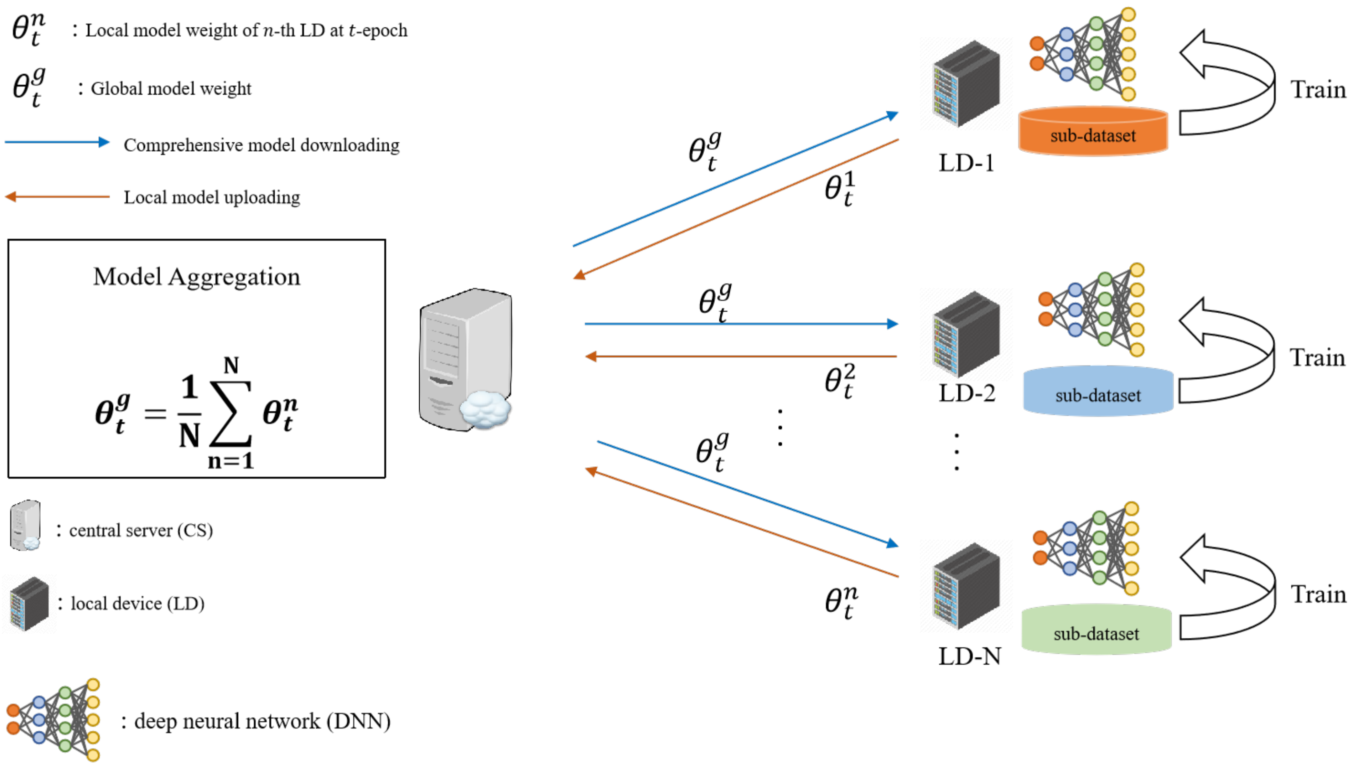


Figure 5. System model of the proposed DecentAMC method.

4.1.1. Broadcasting Initial Comprehensive Model

The CS sets initial parameters and builds initial comprehensive model, and then the CS broadcasts this initial model to all LDs.

4.1.2. Training, Updating and Uploading Local Model

After receiving model and weight, each LD performs local computing and uploads the new model and weight to CS.

4.1.3. Local Models Aggregation

After receiving all local models, CS aggregates these models to obtain a new comprehensive model. The method of aggregation is the key step of DecentAMC. In this paper, we adopt the model averaging (MA) [39], which can be expressed as

$$\theta_t^g = \frac{1}{N}\sum_{n=1}^{N}\theta_t^n, \tag{8}$$

where t represents t-th epoch, N is the number of LDs and θ_t^g denotes aggregated global model weight.

4.1.4. Global Model Downloading

After local model weights are aggregated by CS, CS sends the new global model to all LDs. LDs replace their original model with this new global model, and then repeat step (2), step (3) and step (4) until loss convergence. The algorithm of the proposed ResNet-based DecentAMC is shown in Algorithm 1.

Algorithm 1 Algorithm statement of the proposed ResNet-based DecentAMC method.

Input: IQ samples and corresponding labels.
Output: θ^g_{T-1}.

CS sets initial parameters and builds initial global model (i.e., ResNet) and then send this model to all local devices.

- N: the number of LDs.
- B: the number of batches in each epoch.
- T: the total number of communications, i.e., 100.
- θ^n_t: the single model weight of n-th LS at t-th epoch.
- θ^g_t: the global model weight aggregated by CS at t-th epoch.

for $t = 0, 1, 2, \cdots, T-1$ **do**:
  All LDs download the latest global model weight $\theta^n_{t,b}$.
  for $b = 0, 1, 2, \cdots, B-1$ **do**:
    All LDs train and update local model weight.
  end for
  All LDs upload local model parameter $\{\theta^n_{t,B}\}^N_{n=1}$ to CS.
  CS updates global model parameter by model aggregation

$$\theta^g_t = \frac{1}{N}\sum_{n=1}^{N}\theta^n_{t,B}.$$

end for
return θ^g_{T-1}

5. Simulation Results and Discussions

5.1. Dataset Description

5.1.1. RadioML 2016.10a

In order to prove the superiority of DL-based AMC methods over ML-based AMC methods and the universality of DL, we choose five modulated signals {8PSK, BPSK, QAM16, QAM64 and QPSK} from RadioML 2016.10a [40] to analyze the classification performance of SVM-based CentAMC method and DNN-based CentAMC methods, where the DNNs are mentioned in Section 3.2. As for DNNs, under each SNR, each modulated signal generates 1000 pieces of data, of which 60% are used for training set and the remaining 40% as the test set, and 30% of training set are used for validation set. It must be stressed that SVM and DNNs are all based on centralized method in this simulation. This guarantees the classification performance will not be affected due to the insufficient dataset.

5.1.2. RadioML 2018.01a

We choose a richer dataset, RadioML 2018.01a [41], to analyze the classification performance of the LocalAMC, CentAMC and DecentAMC using different DNNs (CNN, MCNet and ResNet). This dataset includes 24 different modulated signals {16APSK, 32APSK, 64APSK, 128APSK, 16QAM, 32QAM, 64QAM, 128QAM, 256QAM, AM-SSB-SC, AM-DSB-WC, AM-DSB-SC, FM, GMSK, OQPSK, 4ASK, 8ASK, OOK, BPSK, QPSK, 8PSK, 16PSK and 32PSK}, and is generated by GNU Radio. Under each SNR, each modulated signal generates 4096 pieces of data, 75% of the dataset is used for the training set and the rest is used as the test set, and 30% of training set is used as the validation set. In this simulation, we assume that there are 12 LDs and 1 CS. Detailed parameters are shown in Table 3.

5.2. Comparative AMC Methods

5.2.1. AMC Method Based on Local Framework

LocalAMC trains model in LD. This method will achieve the worst classification performance of all deep-learning-based AMC methods because training data are not sufficient and the computational power of a single LD is extremely limited.

5.2.2. AMC Method Based on Centralized Framework

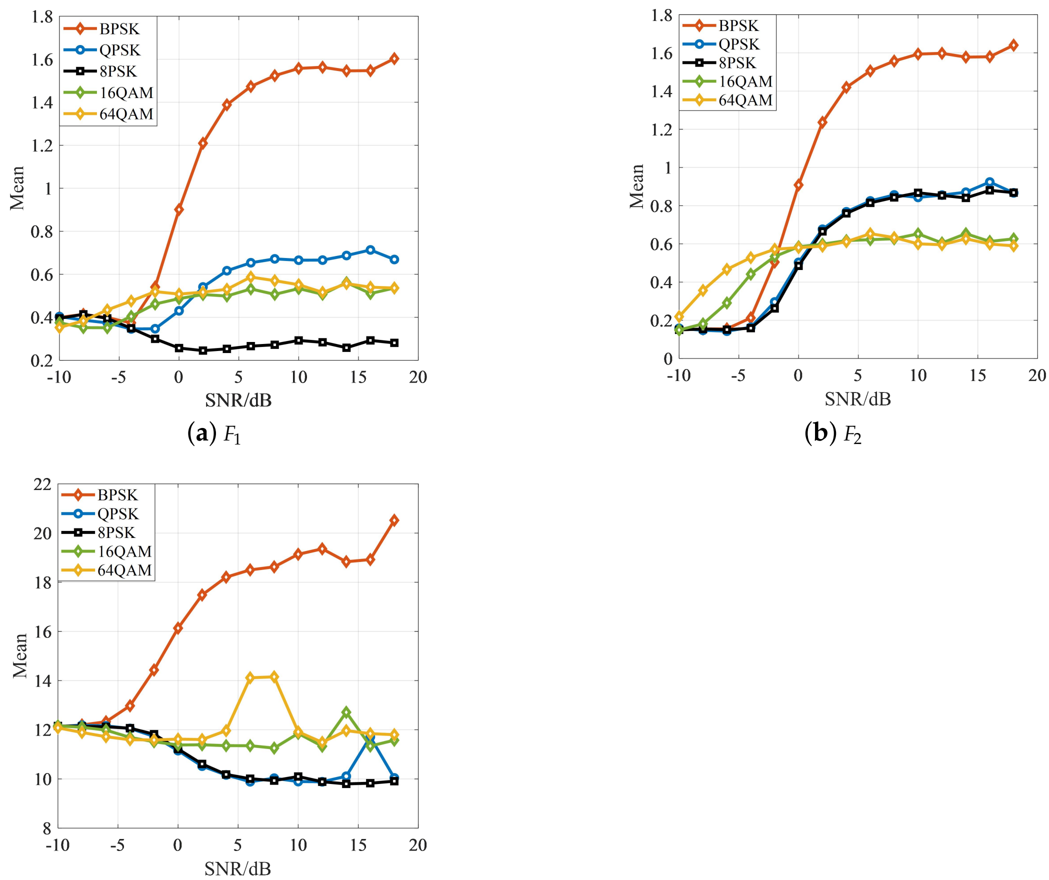
CentAMC trains model in CS based on sufficient dataset collected from all LDs. This method can achieve the best classification performance of all deep-learning-based AMC methods, however the CS will be under great computational pressure in processing a huge amount of data.

Table 3. Simulation parameters for the DNN-based AMC methods.

Parameter	Value
Device	GeForce GTX 2080Ti
Dataset	DeepSig RadioML (version 2018.01A)
Batch size of training	50
Batch size of testing	20
Epoch	100
Learning rate η	0.001
Environment	Keras 2.2.4
Optimizer	Adam

5.3. Classification Performance: ML vs. DL

For AMC based on machine learning, the suitability of artificial feature design will directly affect identification performance. Therefore, we first analyze the designed artificial features on the RadioML 2016.10a dataset. As shown in Figure 6, the selected features can effectively distinguish the five types of digital modulation signals. It also can be seen that, when the signal-to-noise ratio is greater than 5 dB, the separation of five signals is stable and no longer increases with increasing SNR.



(**a**) F_1 (**b**) F_2 (**c**) F_3

Figure 6. The analysis of means of three features of five digital modulation signals under different SNR.

The classification accuracy with varying SNR is considered to represent the classification performance of SVM-based CentAMC and DNN (CNN, MCNet and ResNet)-based CentAMC, which can be expressed as

$$P^i_{Acc} = \frac{N^i_{correct}}{N^i_{test}} \times 100\%, \tag{9}$$

where P^i_{Acc} represents the classification accuracy at SNR = i dB and i range from -10 dB to 18 dB. $N^i_{correct}$ denotes the number of the correctly identified samples at SNR $= i$ dB and N^i_{test} denotes the number of test samples at SNR $= i$ dB. The curves of P^i_{Acc} of SVM-based CentAMC method and DNN-based CentAMC method are shown in Figure 7. It can be observe that three DNNs perform better than SVM. At the same SNR, the accuracy of DNN-based CentAMC method is 3.49%∼9.45% higher than that of SVM-based CentAMC method. The performance of the machine-learning-based AMC and the deep learning-based AMC stabilizes when the SNR is greater than 5 dB, because the separation of five signals is stable and no longer increases with increasing SNR.

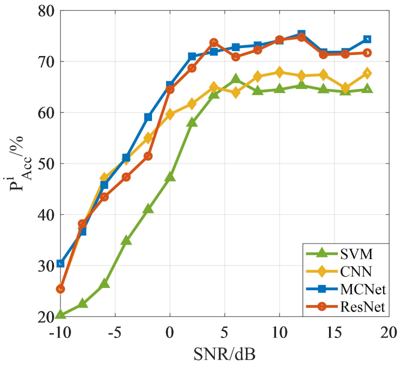


Figure 7. Classification accuracy of P^i_{Acc} SVM-based CentAMC method and DNN (CNN, MCNet and ResNet)-based CentAMC method.

Different number of neural network layers or number of neurons can have an impact on the identification performance, and therefore it is necessary to explain the parameter configuration of the used ResNet in detail. Specifically, we discussed the LocalAMC performance when the number of residual blocks is 1, 2, 3, 4, 7 and 16, respectively. As shown in Figure 8, ResNet with one residual module has the lowest recognition accuracy and ResNet with three residual modules has the highest recognition accuracy, reaching around 90%. Under high SNR conditions, the recognition accuracy of ResNet using 7 residual modules and 16 residual modules actually decreases. Therefore, we design a ResNet with three residual blocks.

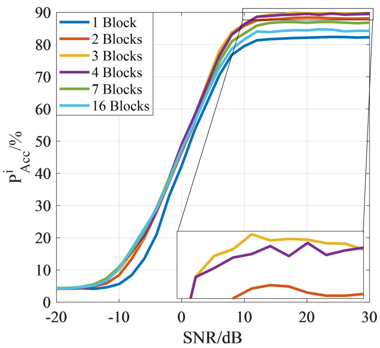


Figure 8. Classification accuracy of LocalAMC based on ResNet with different number of residual blocks on RadioML 2018.01a.

5.4. Classification Performance: Different AMC Methods Based on CNN, MCNet and Proposed ResNet

The Correct Classification Probability under Different SNR

The evaluation criteria for classification performance are the same as Equation (9) mentioned in the Section 5.3. The only significant difference is that the SNR i ranges from -20 dB to 30 dB. The curves of the correct classification probability of LocalAMC, CentAMC and DecentAMC using three DNNs (CNN, MCNet and ResNet) are shown in Figure 9. The classification accuracies P^i_{Acc}, $i \in \{0, 10, 20, 30\}$ dB of DNN (i.e., CNN, MCNet and ResNet)-based AMC methods are shown in Table 4.

It can be observed that, whatever the kind of structure of network, CentAMC and DecentAMC performs significantly better than LocalAMC. As for CNN-based AMC methods and MCNet-based AMC methods, the DecentAMC method has limited performance loss when compared with CentAMC method. Specifically, the performance gap is nearly 2% at high SNR. As for the proposed ResNet-based DecentAMC method in this paper, the error of DecentAMC method and CentAMC method is less than 0.5% and the curves of them almost coincide as shown in Figure 9.

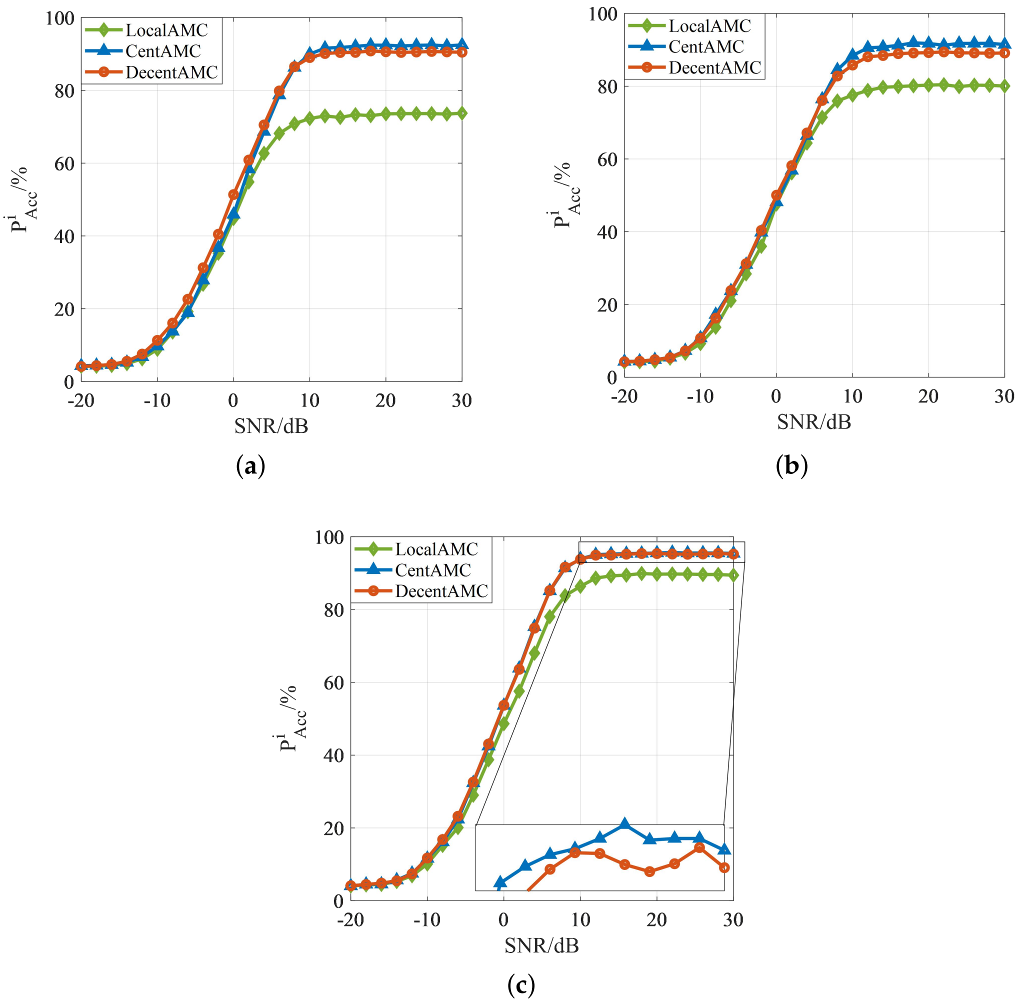


Figure 9. Classification accuracy P^i_{Acc} of DNN-based AMC methods. (**a**) Classification accuracy P^i_{Acc} of CNN-based AMC methods; (**b**) Classification accuracy P^i_{Acc} of MCNet-based AMC methods; (**c**) Classification accuracy P^i_{Acc} of ResNet-based AMC methods.

Table 4. Classification accuracy P^i_{Acc} of DNN-based AMC methods.

(a) Classification Accuracy P^i_{Acc} of CNN-Based AMC Methods				
Method (CNN-Based)	P^0_{Acc}	P^{10}_{Acc}	P^{20}_{Acc}	P^{30}_{Acc}
LocalAMC	44.79%	72.26%	73.55%	73.72%
CentAMC	45.91%	89.96%	92.41%	92.49%
DecentAMC	51.36%	89.00%	90.60%	90.47%
(b) Classification Accuracy P^i_{Acc} MCNet-Based AMC Methods				
Method (MCNet-Based)	P^0_{Acc}	P^{10}_{Acc}	P^{20}_{Acc}	P^{30}_{Acc}
LocalAMC	47.62%	77.56%	80.31%	80.06%
CentAMC	48.17%	88.45%	91.74%	91.46%
DecentAMC	50.04%	85.82%	89.23%	89.13%
(c) Classification Accuracy P^i_{Acc} ResNet-Based AMC Methods				
Method (ResNet-Based)	P^0_{Acc}	P^{10}_{Acc}	P^{20}_{Acc}	P^{30}_{Acc}
LocalAMC	48.65%	86.38%	89.72%	89.45%
CentAMC	53.63%	93.92%	95.54%	95.41%
DecentAMC (proposed)	53.69%	93.80%	95.39%	95.24%

We specially compared the classification accuracy of ResNet-based DecentAMC method with that of CNN-based DecentAMC method and MCNet-based method. As shown in Table 5 and Figure 10, the highest accuracy of DecentAMC based on CNN and MCNet is nearly 90% and the one based on ResNet is over 95%. At high SNR, the accuracy of ResNet-based DecentAMC method is 4.77%∼7.98% higher than that of CNN-based DecentAMC method and MCNet-based DecentAMC method. The accuracy at 30 dB is slightly lower than that at 20dB, no more than 0.2%, which is consistent with the phenomenon in paper [15]. This seemingly abnormal result can actually be explained by the fact that only a limited number of simulation experiments were carried out.

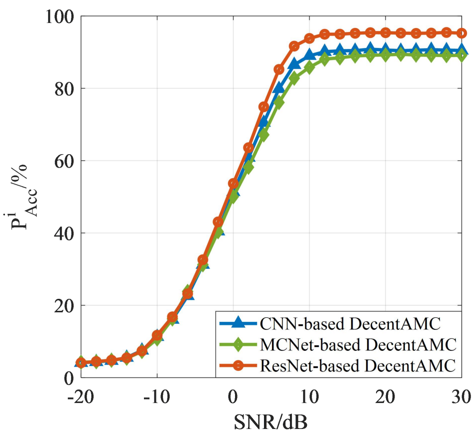


Figure 10. Classification accuracy P^i_{Acc} of DecentAMC method based on CNN, MCNet and ResNet.

Table 5. Classification accuracy P^i_{Acc} of DecentAMC for CNN, MCNet and ResNet.

DecentAMC	P^0_{Acc}	P^{10}_{Acc}	P^{20}_{Acc}	P^{30}_{Acc}
CNN	51.36%	89.00%	90.60%	90.47%
MCNet	50.04%	85.82%	89.23%	89.13%
DecentAMC (proposed)	53.69% (3.65%↑, 2.09%↑)	93.80% (7.98%↑, 4.80%↑)	95.39% (6.16%↑, 4.79%↑)	95.24% (6.11%↑, 4.77%↑)

Note 1: The red font corresponds to the performance improvement of ResNet over MCNet. Note 2: The blue font corresponds to the performance improvement of ResNet over CNN.

5.5. Communication Overhead: ResNet-Based AMC Methods vs. Comparative AMC Methods

One drawback of CentAMC is that each LD needs to upload sub-data to CD, which consumes significant communication resources. DecentAMC chooses to transmit model trained by the sub-data rather than sub-data itself to reduce communication overhead. The model sizes of three DNNs (i.e., CNN, MCNet and ResNet) are shown in Table 6.

Table 6. The model size of three DNNs (i.e., CNN, MCNet and ResNet).

Network Structure	M_s
CNN	678 Kb
MCNet	637 Kb
ResNet	649 Kb

In the LocalAMC method, LDs train model in itself. There is no CS which means communication overhead $O_{Local} = 0$. The CentAMC method includes two times communication. Specifically, one time is each LD uploads its data to the CS, and the other is CS sends final comprehensive model to all LDs. The communication overhead O_{Cent} can be described as [15]

$$O_{Cent} = N(D_s + M_s), \tag{10}$$

where the N represents the number of LDs, and the D_s denotes the data size of each LD, and the M_s means the comprehensive model size. In this simulation, the D_s is 1,431,738 Kb. In the DecentAMC method, there are multiple times communication between CS and LDs, because from the point view of LD, there exists frequent uploading of the local model and frequent downloading of the global model. The communication overhead O_{Decent} can be written as

$$O_{Decent} = 2NM_sT + NM_s, \tag{11}$$

where the T represents the total communication times.

The communication overhead of LocalAMC, CentAMC and DecentAMC using three DNNs (CNN, MCNet and ResNet) is shown in Table 7. It can be observed that, no matter what kind of network, the communication overhead of DecentAMC is much lower than that of CentAMC, and the exact percentage declines are approximately 90.49%, 91.06% and 90.89% in the case of CNN, MCNet and ResNet, respectively. For the DecentAMC, there is no significant difference in the communication overhead of the three DNN structures due to the close size of their network models. To further improve communication efficiency, we will explore how to combine Elastic averaging SGD [42] with AMC in the future.

Table 7. The communication overhead of LocalAMC, CentAMC and DecentAMC based on three DNNs (CNN, MCNet and ResNet).

Network Structure	O_{Local}	O_{Cent}	O_{Decent}
CNN	0 Kb	17,188,992 Kb	1,635,336 Kb (90.49%↓)
MCNet	0 Kb	17,188,500 Kb	1,536,444 Kb (91.06%↓)
ResNet	0 Kb	17,188,644 Kb	1,565,388 Kb (90.89%↓)

Note: The red font indicates the percentage of communication overhead reduced by DecentAMC compared to CentAMC.

6. Conclusions

In this paper, we verified the superiority of DL over ML by utilizing CNN, MCNet, ResNet and SVM in the drone communication systems. Then, we used these three DNNs to compare the classification performance of three different training methods: the LocalAMC, the CentAMC and the DecentAMC. DecentAMC based on CNN and MCNet has similar classification performance to CentAMC while protecting data privacy and reducing communication overhead. The classification performance of ResNet-based DcentAMC is quite close to that of

ResNet-based CentAMC. Last but not least, we proved that the proposed ResNet-based DecentAMC performs better than the other two DNN (CNN and MCNet)-based DecentAMC in the drone communication systems. In future work, we shall consider the algorithm deployment in the real drone communication systems to realize the signal recognition.

Author Contributions: Conceptualization, M.M., Y.X. and Z.W.; methodology, Y.X.; software, X.F.; writing—original draft preparation, M.M.; writing—review and editing, G.G. and X.F.; visualization, Y.X.; funding acquisition, M.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received was supported in part by the Jiangsu University's Blue Project Funding under grant number 2022, by the Natural Science Foundation Project of Nanjing Vocational College of Information Technology (Special Key for Vice Senior High School) under grant number YK20210501.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: We are sorry that the dataset of this article cannot be provided due to privacy and moral restrictions.

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

References

1. Qi, P.; Zhou, X.; Ding, Y.; Zhang, Z.; Zheng, S.; Li, Z. FedBKD: Heterogenous federated learning via bidirectional knowledge distillation for modulation classification in IoT-edge system. *IEEE J. Sel. Top. Signal Process.* **2022**, *17*, 189–204. [CrossRef]
2. Dong, B.; Liu, Y.; Gui, G.; Fu, X.; Dong, H.; Adebisi, B.; Gacanin, H.; Sari, H. A lightweight decentralized learning-based automatic modulation classification method for resource-constrained edge devices. *IEEE Internet Things J.* **2020**, *9*, 24708–24720. [CrossRef]
3. Hou, C.B.; Liu, G.W.; Tian, Q.; Zhou, Z.C.; Hua, L.J.; Lin, Y. Multi-signal modulation classification using sliding window detection and complex convolutional network in frequency domain. *IEEE Internet Things J.* **2022**, *9*, 19438–19449. [CrossRef]
4. Yang, J.; Gu, H.; Hu, C.; Zhang, X.; Gui, G.; Gacanin, H. Deep complex-valued convolutional neural network for drone recognition based on RF fingerprinting. *Drones* **2022**, *6*, 374. [CrossRef]
5. 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]
6. Mao, Q.; Hu, F.; Hao, Q. Deep learning for intelligent wireless networks: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 2585–2621. [CrossRef]
7. Eldemerdash, Y.A.; Dobre, O.A.; Öner, M. Signal identification for multiple-antenna wireless systems: Achievements and challenges. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 1524–1551. [CrossRef]
8. Dobre, O.A. Signal identification for emerging intelligent radios: Classical problems and new challenges. *IEEE Instrum. Meas. Mag.* **2015**, *18*, 11–18. [CrossRef]
9. Zhu, Z.; Nandi, A. *Automatic Modulation Classification: Principles, Algorithms and Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
10. Kim, K.; Polydoros, A. Digital modulation classification: The BPSK versus QPSK case. In Proceedings of the IEEE Military Communications Conference (MILCOM), San Diego, CA, USA, 23–26 October 1988; pp. 431–436.
11. Freund, Y.; Schapire, R.E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
12. Panagiotou, P.; Anastasopoulos, A.; Polydoros, A. Likelihood ratio tests for modulation classification. In Proceedings of the IEEE Military Communications Conference (MILCOM), Los Angeles, CA, USA, 22–25 October 2000; pp. 670–674.
13. Huang, S.; Dai, R.; Huang, J.; Yao, Y.; Gao, Y.; Ning, F.; Feng, Z. Automatic modulation classification using gated recurrent residual network. *IEEE Internet Things J.* **2020**, *7*, 7795–7807. [CrossRef]
14. Chang, S.; Huang, S.; Zhang, R.; Feng, Z.; Liu, L. Multitask-learning-based deep neural network for automatic modulation classification. *IEEE Internet Things J.* **2022**, *9*, 2192–2206. [CrossRef]
15. Fu, X.; Gui, G.; Wang, Y.; Gacanin, H.; Adachi, F. Automatic modulation classification based on decentralized learning and ensemble learning. *IEEE Trans. Veh. Technol.* **2022**, *71*, 7942–7946. [CrossRef]
16. Nandi, A.K.; Azzouz, E.E. Algorithms for automatic modulation recognition of communication signals. *IEEE Trans. Commun.* **1998**, *46*, 431–436. [CrossRef]
17. Wang, F.; Wang, Y.; Chen, X. Graphic constellations and DBN based automatic modulation classification. In Proceedings of the IEEE Vehicular Technology Conference (VTC Spring), Sydney, NSW, Australia, 4–7 June 2017; pp. 1–5.

18. Wang, L.; Guo, S.; Jia, C. Recognition of digital modulation signals based on wavelet amplitude difference. In Proceedings of the IEEE International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 26–28 August 2016; pp. 627–630.
19. Gardner, W. Measurement of spectral correlation. *IEEE Trans. Acoust. Speech Signal Process.* **1986**, *34*, 1111–1123. [CrossRef]
20. Gardner, W.A.; Spooner, C.M. Cyclic spectral analysis for signal detection and modulation recognition. In Proceedings of the IEEE Military Communications Conference (MILCOM), San Diego, CA, USA, 23–26 October 1988; pp. 419–424.
21. Liu, M.; Song, T.; Hu, J.; Yang, J.; Gui, G. Deep learning-inspired message passing algorithm for efficient resource allocation in cognitive radio networks. *IEEE Trans. Veh. Technol.* **2019**, *68*, 641–653. [CrossRef]
22. Tan, J.; Zhang, L.; Liang, Y.-C.; Niyato, D. Intelligent sharing for LTE and WiFi systems in unlicensed bands: A deep reinforcement learning approach. *IEEE Trans. Commun.* **2020**, *68*, 2793–2808. [CrossRef]
23. Huang, H.; Peng, Y.; Yang, J.; Xia, W.; Gui, G. Fast beamforming design via deep learning. *IEEE Trans. Veh. Technol.* **2020**, *69*, 1065–1069. [CrossRef]
24. O'Shea, T.; Hoydis, J. An introduction to deep learning for the physical layer. *IEEE Trans. Cogn. Commun. Netw.* **2017**, *3*, 563–575. [CrossRef]
25. Lin, Y.; Tu, Y.; Dou, Z. An improved neural network pruning technology for automatic modulation classification in edge devices. *IEEE Trans. Veh. Technol.* **2020**, *69*, 5703–5706. [CrossRef]
26. Huang, S.; Chai, L.; Li, Z.; Zhang, D.; Yao, Y.; Zhang, Y.; Feng, Z. Automatic Modulation Classification Using Compressive Convolutional Neural Network. *IEEE Access* **2019**, *7*, 79636–79643. [CrossRef]
27. Zhang, Z.; Wang, C.; Gan, C.; Sun, S.; Wang, M. Automatic modulation classification using convolutional neural network with features fusion of SPWVD and BJD. *IEEE Trans. Signal Inf. Process. Netw.* **2019**, *5*, 469–478. [CrossRef]
28. Wang, Y.; Yang, J.; Liu, M.; Gui, G. LightAMC: Lightweight automatic modulation classification via deep learning and compressive sensing. *IEEE Trans. Veh. Technol.* **2020**, *69*, 3491–3495. [CrossRef]
29. Meng, F.; Chen, P.; Wu, L.; Wang, X. Automatic modulation classification: A deep learning enabled approach. *IEEE Trans. Veh. Technol.* **2018**, *67*, 10760–10772. [CrossRef]
30. Cao, Z.; Shih, W.-T.; Guo, J.; Wen, C.-K.; Jin, S. Lightweight convolutional neural networks for CSI feedback in massive MIMO. *IEEE Commun. Lett.* **2021**, *25*, 2624–2628 . [CrossRef]
31. Qi, P.; Zhou, X.; Zheng, S.; Li, Z. Automatic modulation classification based on deep residual networks with multimodal information. *IEEE Trans. Cogn. Commun. Netw.* **2021**, *7*, 21–33. [CrossRef]
32. Bal, H.E.; Pal, A. Parallel and distributed machine learning algorithms for scalable big data analytics. *Future Gener. Comput. Syst.* **2020**, *108*, 1159–1161. [CrossRef]
33. Yang, Z.; Bajwa, W.U. ByRDiE: Byzantine-resilient distributed coordinate descent for decentralized learning. *IEEE Trans. Signal Inf. Process. Netw.* **2019**, *5*, 611–627. [CrossRef]
34. Wang, Y.; Guo, L.; Zhao, Y.; Yang, J.; Adebisi, B.; Gacanin, H.; Gui, G. Distributed learning for automatic modulation classification in edge devices. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 2177–2181. [CrossRef]
35. Fu, X.; Gui, G.; Wang, Y.; Ohtsuki, T.; Adebisi, B.; Gacanin, H.; Adachi, F. Lightweight network and model aggregation for automatic modulation classification in wireless communications. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 29 March–1 April 2021; pp. 1–6.
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp.770–778.
37. Swami, A.; Sadler, B.M. Hierarchical digital modulation classification using cumulants. *IEEE Trans. Commun.* **2000**, *48*, 416–429 . [CrossRef]
38. Huynh-The, T.; Hua, C.; Pham, Q.; Kim, D. MCNet: An efficient CNN architecture for robust automatic modulation classification. *IEEE Commun. Lett.* **2020**, *24*, 811–815. [CrossRef]
39. McDonald, R.; Hall, K.; Mann, G. Distributed training strategies for the structured perceptron. In Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics, Los Angeles, CA, USA, 2 June 2010; pp. 456–464.
40. Oshea, T.J.; West, N. Radio machine learning dataset generation with GNU radio. In Proceedings of the GNU Radio Conference, Boulder, CO, USA, 6 September 2016; pp. 1–6.
41. 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]
42. Zhang, S.; Choromanska, A.E.; LeCun, Y. Deep learning with elastic averaging SGD. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1–9.