Generalized Automatic Modulation Classification Under Non-Gaussian Noise with Varying SNR Conditions: A CNN Enable Method

Yu Wang, *Student Member, IEEE*, Guan Gui, *Senior Member, IEEE*, Tomoaki Ohtsuki, *Senior Member, IEEE*, Fumiyuki Adachi, *Life Fellow, IEEE*

Abstract—Automatic modulation classification (AMC) is an critical step to identify signal modulation types so as to enable more accurate demodulation in the non-cooperative scenarios. Convolutional neural network (CNN)-based AMC is believed as one of the most promising methods with great classification accuracy. However, the conventional CNN-based methods are lack of generality capabilities under time-varying signal-to-noise ratio (SNR) conditions, because these methods are merely trained on specific datasets and can only work at the corresponding condition. In this paper, a novel CNN-based generalized AMC method is proposed, and a more realistic scenario is considered, including white non-Gaussian noise and synchronization error. Its generalization capability stems from the mixed datasets under varying noise scenarios, and the CNN can extract common features from these datasets. Simulation results show that our proposed architecture can achieve higher robustness and generalization than the conventional ones.

Index Terms—Automatic modulation classification, convolutional neural network, generalization, white non-Gaussian noise, synchronization error.

I. Introduction

Automatic modulation classification (AMC) is an essential technology in non-cooperative communication systems for demodulation tasks of unknown signals [1]–[4]. It has various applications, such as intercepted enemy signal recovery, adaptive modulator [5], and spectrum sensing [6], in both military and civilian strategies. In recent years, various methods were proposed for AMC, and they can be classified into two common AMC methods that are based on likelihood functions and features [7], respectively.

In the likelihood-based methods, AMC can be formulated as a hypothesis testing problem [8]. It is necessary to design a correct likelihood function to evaluate likelihood for each modulation type within hypothesis pool. Then, the likelihoods of each modulation type are compared to make a final decision. However, the likelihood-based AMC methods excessively depend on channel state information (CSI) of wireless channels.

In the feature-based methods, AMC is modeled as a pattern recognition problem and it consists of three steps: pre-processing, feature extraction, and classifier design [9]. Various AMC methods have been developed using instantaneous features (or signal spectral-based features), wavelet transform-based features, high-order statistics-based features, cyclic spectrum analysis-based features, and so on. To realize modulation type classification by extracted features, they usually adopt classifiers, such as support vector machine (SVM), decision tree (DT), k-nearest neighbor (KNN) and multilayer perceptron (MLP).

In recent years, deep learning (DL) is considered as a powerful tool, because it is expert in automatic feature extraction from huge amounts of data, instead of the complex and difficult design of manmade features [10], [11]. For this reason, DL has been successfully applied in wireless communications [12]–[16] and Internet-of-Things [18]–[24].

In addition, DL has been applied in multiple-input and multiple-output (MIMO) [17], non-orthogonal multiple-access (NOMA), and cognitive radio (CR). For example, H. Huang, et al. [25], [26] proposed a fast beam forming technology for downlink MIMO based on unsupervised learning. G. Gui, et al. [27] applied a long short-term memory (LSTM) network into a typical NOMA system for enhancing spectral efficiency. M. Liu, et al. [28], [29] introduced DL into resource allocation in CR.

Moreover, state-of-the-art DL-based AMC methods have been developed in recent two years. T. OShea and J. Hoydis [30] proposed a convolutional neural network (CNN)-based AMC, which is realized by training CNN on the in-phase and quadrature (IQ) components of signals. B. Tang, et al. [31] transformed the modulated signals into constellation diagrams, and then generative adversarial network(GAN) was applied to distinguish these constellation diagrams. Y. Tu, et al. [33] proposed a lightweight and fast CNN-based AMC method for edge devices on their previous works [32], [33]. Y. Wang, et al. [34] proposed combined IQ sample-based CNN and constellation diagram-based CNN method to recognize different modulation types.

Although these CNN-based AMC methods have been proposed to demonstrate better performance than traditional meth-

Y. Wang and G. Gui are with the College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (E-mails: {1018010407, guiguan}@njupt.edu.cn).

T. Ohtsuki is with the Department of Information and Computer Science, Keio University, Yokohama 223-8521, Japan (E-mail: ohtsuki@ics.keio.ac.jp).

F. Adachi is with the Research Organization of Electrical Communication (ROEC), Tohoku University, Sendai 980-8577 Japan (E-mail: adachi@ecei.tohoku.ac.jp)
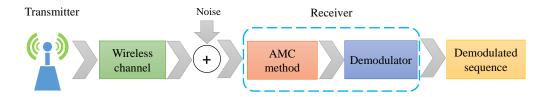
Fig. 1. The AMC-based receiver in a non-cooperative communication system. There are no agreement and authentication between receivers and transmitters in this system. The demodulation relies on the "AMC method" module to correctly and fast identify modulation types, and then it has possibility to demodulate the received signals.

ods, most of them are trained by dataset with single signal-to-noise ratio (SNR). It means that these CNNs can just achieve satisfying performance at the corresponding single SNR rather than all SNR scenarios. These independent CNNs are hard to be generalized. If we adopt these CNNs in practical applications, we must train various CNNs with dataset collected from different SNR conditions, and choose correct CNN-models according to actual communication environments, which is not convenient.

In this paper, a CNN-based robust generalized AMC method with higher generality capability under varying noise conditions is proposed, considering a more practical scenario, characterized by white non-Gaussian noise (WNGN) and non-ideal synchronization, i. e., frequency offset or phase offset. The proposed AMC method has more powerful recognition capability than the traditional feature-based AMC method, and it achieves higher robustness for actual applications at a slight performance loss. Compared with other CNN-based AMC methods, our proposed method has two obvious advantages, which are listed as follows:

Undesired time-varying SNR estimation: Other CNN-based AMC methods rely on a precise SNR estimation, because CNNs are trained on samples with single SNR. For multiple CNNs-based solution [7], [9], [30], [32]–[34], a precise SNR estimation is essential to assist systems in selecting correct CNN model from trained models. If SNR cannot be estimated precisely, these methods may be ineffective. Unlike these conventional methods, CNN in our proposed method is trained on a mixed dataset containing different signals with SNR $\in$ {-5 dB, 0 dB, 5 dB}. Unknown signals with SNR ranging from -5 dB to 5 dB can be recognized by the same CNN, and SNR estimation is unwanted.

Less device memory: When SNR is ranging from -5 dB to 5 dB with an interval 1 dB, far more than one CNN model should be trained for responding to different SNR conditions in other CNN-based AMC methods. However, our proposed method just need to train one CNN model in actual applications. In the case of the same network structures, our method just requires less device memory than other CNN-based AMC methods.

The rest of this paper is organized as follows. Section II includes the system model, the signal model, and dataset. In Section III, we propose a CNN-based generalized AMC (GAMC) methods. In Section IV, various simulation results are provided to compare their performance, respectively. Finally, we conclude this paper.

II. SYSTEM MODEL AND SIGNAL MODEL

A. System Model

A typical non-cooperative communication system is considered, where transmitters transmit digital modulation signals through wireless channel, and the receiver does not get a priori information about modulation types, symbol rates, and so on.

After receiving these digital modulation signals, the system makes preprocessing, including down conversion, low pass filtering, and analog-to-digital conversion and so on. After preprocessing, we can get baseband signals, which are fed into an AMC module to identify modulation types. The AMC-based receiver is shown in Fig. 1.

In this paper, we focus on the feature-based AMC methods, and they generally consist of three steps: processing, feature extraction, and classification [9]. In traditional AMC methods, the most difficult part is the design of effective manmade features, and classifiers are usually based on machine learning or simple threshold detection, which is shown in Fig. 2(a).

Unlike the traditional methods, DL methods, e. g., CNN or recurrent neural network (RNN), can simultaneously achieve feature extraction and classification. Moreover, the DL-based AMC method can get rid of complex and difficult manmade feature design. The framework of the DL-based AMC method is depicted as Fig. 2(b).
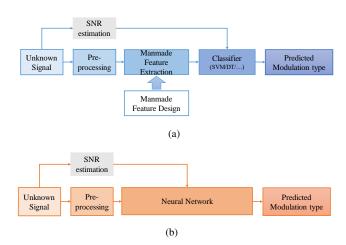

Fig. 2. Two feature-based AMC methods: (a) The structure of a traditional AMC method. (b) The structure of a DL-based AMC method. The modules "Manmade Feature Extraction" and "Classifier" in the traditional AMC method is replaced by the neural network in DL-based AMC method, and the DL-based AMC method can throw away the complex and difficult step of the manmade feature design, because DL algorithms have the powerful capabilities to automatically extract effective features.

B. Signal Model

Assuming that the received complex-valued baseband equivalent signal $r(n)$ is sampled at the Nyquist rate, it is given as follows.

$$r(n) = \alpha e^{j(\Delta\theta + 2\pi\Delta f \frac{n}{N})} x(n) + w(n), n = 1, 2, ..., N, \quad (1)$$

where α, $\Delta\theta$, Δf, and N represent attenuation factor, carrier phase offset (CPO) caused by wireless channels [7], normalized carrier frequency offset (CFO), and the number of sampling points in an independent observation phase, respectively. In this paper, we consider a flat fading and time invariant channel, so α, $\Delta\theta$, and Δf are constant in each observation phase.

In addition, $\{x(n)\}_{n=1}^N$ is the symbol sequence, and three modulation types of frequency shift keying (FSK), phase shift keying (PSK), and quadrature amplitude modulation (QAM) are considered in this paper; $w(n)$ represents additive noise. In this paper, we mainly consider WNGN based on Gaussian mixture model (GMM) with K components [35]. GMM-based WNGN consists of K independent noise obeying complex Gaussian distributions. Its probability density function (PDF) is given as

$$f(w(n)) = \sum_{k=1}^{K} \frac{\lambda_k}{\sqrt{\pi\sigma_k^2}} e^{\frac{-|w(n)|^2}{\sigma_k^2}}, \quad (2)$$

and it consists of K independent noise obeying complex Gaussian distributions. λ_k is the ratio of each component, and $\sum_{k=1}^{K} \lambda_k = 1$ and $0 < \lambda_k < 1$.

In this paper, we consider a classical GMM-based WNGN with two components of $w_0(n)$ and $w_1(n)$, and $w(n) = \lambda_0 w_0(n) + \lambda_1 w_1(n)$. $w_0(n)$ is just additive white Gaussian noise (AWGN), i. e., $w_1(n) \sim CN(0, \sigma_0^2)$. $w_1(n)$ is impulsive noise with zero mean and the variance of σ_1^2, where $\sigma_1 \gg \sigma_0$ [36].

III. AMC METHODS

In this section, a CNN-based generalized AMC (GAMC) method is proposed with better generalization performance under varying noise conditions. In contrast, previous proposed CNN-based AMC method [7], [9], [30], [32]–[34] is denoted as a fixed AMC (FAMC) method, and it does not equip with powerful generalization capability. In addition, traditional AMC methods, based on classical manmade features and typical machine learning-based classifiers, are firstly introduced as a comparison.

A. Traditional AMC Method

For the purpose of highlighting the performance of DL-based AMC method, we adopt one of a traditional AMC method as a comparison. The structure of the traditional AMC method is shown in Fig. 2(a), where high-order cumulants (HOC) [37] are classical manmade features, which are described below.

In detail, the normalized fourth-order cumulants [38] are applied, and they can be describe as:

$$\widetilde{C}_{40} = \widetilde{M}_{40} - 3\widetilde{M}_{20}^2, \quad (3)$$

$$\widetilde{C}_{41} = \widetilde{M}_{41} - 3\widetilde{M}_{21}\widetilde{M}_{20}, \quad (4)$$

$$\widetilde{C}_{42} = \widetilde{M}_{42} - \left|\widetilde{M}_{20}\right|^2 - 2\widetilde{M}_{21}^2, \quad (5)$$

where $\widetilde{M}_{mk}$ represents the normalized moments. It can be denoted as $\widetilde{M}_{mk} = \frac{\sum_{n=1}^{N} r^{m-k}(n) r^{*k}(n)}{(\sum_{n=1}^{N} |r(n)|^2)^{m/2}}$, and $r^*(n)$ is the conjugate of $r(n)$.

Hence, $\left\{\widetilde{C}_{40}, \widetilde{C}_{41}, \widetilde{C}_{42}\right\}$ works as a feature vector and SVM acts as a clssifier.

B. CNN-based GAMC Method

Here, a CNN-based GAMC method is stated from dataset, CNN structure, classifier, loss function, and training and test strategies.

1) Dataset: Dataset, applied in this paper, contains multiple in-phase and quadrature (IQ) samples, which is transformed from the received complex signal sequence. The received sequence is defined as $R = \{r(n)\}_{n=1}^N$. To avoid the scaling problem, the power of R should be normalized, and the normalized received sequence $\widetilde{R}$ is equal to $\frac{R}{\sum_{n=1}^{N} |r(n)|^2}$.

Then, we separate real part $real\left(\widetilde{R}\right)$ and imaginary part $imag\left(\widetilde{R}\right)$ from the normalized received sequence. Next, $real\left(\widetilde{R}\right)$ and $imag\left(\widetilde{R}\right)$ are combined into a matrix with dimension $2 \times N$, which is treated as one sample for training or testing.

In addition, the real part and imaginary part are also in-phase (I) component and quadrature (Q) component of signal, respectively. So this training or test sample is also called as IQ sample, which is shown in eq. (6).

$$IQ = \left[\begin{array}{c} real(\widetilde{R}) \\ imag(\widetilde{R}) \end{array} \right]. \quad (6)$$

2) CNN structure: The CNN consists of four main parts: "Input", "Output", "Convolutional layer", and "Fully-connected layer", and its structure is depicted in Fig. 3.

Firstly, "Input" is the dataset with IQ samples and their corresponding labels. Then, "Convolutional Layer" is to automatically extract features and contains two "Conv2D" layers. In these "Conv2D" layers, the convolutional kernel sizes are 2×4 and 1×8, respectively, which are is designed according to input data: IQ sample, the dimensionality of which is $2 \times N$.

Next, "Fully-connected Layer" is fundamentally a classifier with three "Dense" layers, and its output is a probability distribution, which contains the possibility of each modulation type. Finally, "Output" is the predicted modulation type, and it is given by the maximum a posteriori (MAP) classifier based on probability distribution of the last "Dense" layer.

Besides, rectified linear unit (ReLU) plays the role of the activation function in each layer except the last dense layer, where Softmax is applied. Assuming that x_i is the output of
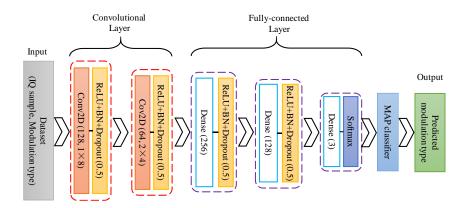
Fig. 3. The CNN structure with four parts: 1) Input: IQ samples and corresponding modulation types as labels; 2) Convolutional Layer: two Conv2D layers for feature extraction; 3) Fully-connected Layer: three Dense layers to give predicted probability distribution; 4) Output: a MAP classifier to classify modulation types according to the predicted probability distribution.

the i-th neuron in a certain layer, the function of ReLU and Softmax can be described as follows.

$$f_{ReLU}(x_i) = \max(0, x_i), f_{Softmax}(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}. \quad (7)$$

In addition, batch normalization (BN) and dropout, after each activation function (except in the last fully-connected layer), are applied to accelerate training, improve performance slightly and avoid overfitting. They can be considered as two implicit regularization terms. BN is to normalize the output of each layer in each batch and can be represented as

$$BN(O_i) = \gamma \cdot \frac{O_i - Mean(O_{minibacth})}{\sqrt{Var(O_{minibacth}) + \epsilon}} + \beta, \quad (8)$$

where $Mean(O_{minibacth})$ and $Var(O_{minibacth})$ are the mean and variance of the output of mini-batch data, respectively; γ and β are trainable parameters [39], and ϵ is a minimum value to prevent denominators from being zero. In addition, Dropout is to temporarily disable partial neurons with a certain probability in the training process.

It is noted that the same CNN structure is applied into both FAMC and GAMC.

3) MAP classifier: AMC is to identify modulation types in a limited modulation type candidate pool. Assuming that the modulation type candidate pool is $M = \{m_i\}_{i=1}^{N_{type}}$, where m_i represents a certain modulation type and this pool contains N_{type} different modulation types, MAP criterion in CNN-based AMC [36] can be described as:

$$\hat{m}_i = \arg\max_{m_i \in M} p(m_i, f_{model}, \Theta | IQ), \quad (9)$$

where $\hat{m}_i$ is the predicted modulation type; f_{model} represents model structure, and $\Theta = \{\Theta_{trainable}, \Theta_{untrainable}\}$ is the model parameters, which contains massive trainable parameters and a few untrainable parameters; $p(\cdot)$ is a PDF that refers to the output of Softmax function in the last fully-connected layer.

4) Loss function: In this paper, the categorical cross entropy (CCE) function is applied as data loss function (or experience loss function), considering that AMC is essentially a multi-class classification task.

Suppose the dataset $T = \{(s_i, l_i)\}_{i=1}^{N_s}$ is applied for the training of CNN, where s_i, l_i and N_s represents IQ sample, ground truth sample label through one-hot encoding, and the number of training samples, respectively. The CCE loss function is given as follows.

$$L_{CCE}(T; f_{model}, \Theta) = -\sum_{i=1}^{N_s} l_i \log(f_{model}(s_i; \Theta)). \quad (10)$$

However, the final loss function contains not only the data loss function but also the structure loss function, and it can be written as follows.

$$L(T; f_{model}, \Theta) = \frac{1}{N_s} L_{CCE}(T; f_{model}, \Theta) \\ + \lambda_{model} J(f_{model}, \Theta), \quad (11)$$

where $J(\cdot)$ is just the structure loss function (or regularization term), which is to avoid overfitting, and λ_{model} is applied to balance these two loss functions. In this paper, BN and dropout are as the components of the structure loss function, which have been introduced above.

5) Training and test strategies: Training optimizer, training process, and test process are introduced in this part. The same training optimizer is adopted in both GAMC and FAMC method, and their main difference focuses on the training and test process between our proposed GAMC method and FAMC method.

Training optimizer: stochastic gradient descent (SGD) is introduced as an optimizer to minimize the function (11) by iteratively optimizing and updating trainable parameters $\Theta_{trainable}$ in Θ. The optimizing rule is written as follows.

$$\Theta_{trainable}^{new} = \Theta_{trainable}^{now} \\ - \eta \frac{\partial L(T; f_{model}, \Theta_{trainable}^{now}, \Theta_{untrainable})}{\partial \Theta_{trainbale}^{now}}, \quad (12)$$

where η is referred to as a learning rate to control scale of parameter adjustment.

Training process: The training processes of GAMC and FAMC are shown in Fig. 4. From Fig. 4(a) and Fig. 4(b), it can be observed that GAMC differs from FAMC in the

IQ dataset for training. CNN in GAMC is trianed on mixed dataset, while CNN in FAMC is fed with single dataset for training.

When training CNN in GAMC, three datasets with SNRs of -5 dB, 0 dB and 5 dB are proportionally mixed. Then, the mixed dataset is divided into training dataset and validation dataset by 7:3 in random. Training dataset is fed into CNN for training and validation dataset is applied to measure the performance of trained CNN after each epoch. The trained CNN can be employed into modulation type recognition of unknown signals with SNR ranging from -5 dB to 5 dB. However, CNN in FAMC is trained on IQ samples with fixed SNR = i dB, and this CNN just can be employed to identify received signals with SNR = i dB.

Powerful generalization performances of GAMC under varying SNR conditions originate from the mixed datasets. CNN in GAMC can extract features from various dataset with different SNRs simultaneously. Then, CNN has the ability to filter out common or similar features to classify modulation types. Hence, these extracted features are universally suitable and more roust under varying noise conditions.

For training GAMC, we choose IQ dataset with different SNRs by sampling SNRs = [-5, 5] dB at an equal interval, i. e., datasets' SNRs = {-5 dB, 0 dB and 5 dB}, and this approach can reduce the amount of dataset used for training.
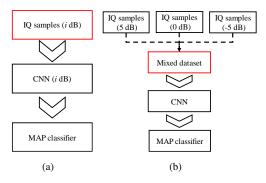

Fig. 4. Training Processes: (a) FAMC; (b) GAMC. FAMC is trained on the dataset with a single SNR, and GAMC is fed with a mixed dataset. Specifically, "CNN (i dB)" in FAMC is trained on "IQ sample (i dB)", and there are eleven CNN models that must be independently trained on their own datasets in FAMC. However, GAMC is trained on the dataset containing three IQ samples with SNRs = {-5, 0, 5} dB, and we just need to train one CNN model and the model can be confronted with arbitrary IQ samples with SNRs ranging from -5 dB to 5 dB.

Test process: In the test process, signal samples in test dataset are fed into the trained CNN. Then CNN will give a predicted probability distribution, and the MAP classifier is applied to identify modulation type. In the test phase, CNN just contains feed forward propagation.

There is a great deal of differences between test processes of GAMC and FAMC, and their test processes are shown in Fig. 5. In the FAMC-aided communication system, multiple CNN models are trained on different IQ samples with fixed SNRs. Hence, it is fundamental to equip with the SNR estimation for the choice of a correct CNN model under varying SNR conditions.

From the test process, the differences between FAMC and GAMC can be easily observed from two aspects. On the one

hand, the SNR estimation technique is removed in GAMC-based system, because there is just one CNN model and GAMC is independent of the SNR estimation technique to obtain the estimated SNR for the next CNN model choice. However, FAMC contains multiple CNN models, in order to confront the varying SNR conditions, and it rely on the estimated SNR to choose a corresponding and correct model.

On the other hand, for the testing of IQ samples with SNRs = [-5, 5] dB, FAMC must prepare eleven CNN models and GAMC just needs one CNN models. It means that GAMC only requires 1/11 device memories of FAMC, and the size of CNN model in GAMC is 23.3 MB, while that in FAMC exceeds 256 MB. Moreover, it is noted that FAMC and GAMC has the same computation complexity, because of the same CNN structure in these two AMC methods.
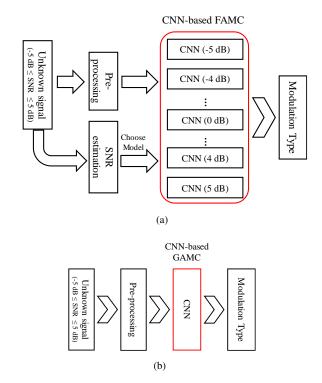

Fig. 5. Test processes: (a) FAMC; (b) GAMC. Training process of FAMC is simpler than that of GAMC, however, it is on the contrary in their test processes. The GAMC's test process is extremely simple, and the test results can be given by CNN, after inputting pre-processed samples without other operations. However, pre-processing and SNR estimation should be implemented at the same time in FAMC, and the latter operation is applied to choose a correct model, such as "CNN (j dB)", from multiple CNN models. Then, the pre-processed IQ samples are fed into "CNN (j dB)" to give the predicted modulation types.

IV. EXPERIMENTAL RESULTS

In this section, simulation results will be given to evaluate performances of various AMC methods. The detailed parameters and their corresponding values for simulation are given in Table I. It is noted that the amount of training samples in GAMC is 20000 samples per type per SNR, but GAMC applies three datasets for training, and there are 60000 samples fed into CNN in total. Thus, for fair comparison, 60000 samples per type per SNR are created for the training of CNN in FAMC.

Parameter	Value
Modulation type candidate pool	$M=\{\text{FSK, PSK, QAM}\}$, $N_{type} = 3$
WNGN $(\lambda_0, \lambda_1, \sigma_0, \sigma_1)$	$\lambda_0 = 0.9, \lambda_1 = 0.1$, $\sigma_1^2/\sigma_0^2 = 100$
CPO $\Delta\theta$	$\Delta\theta \sim U(0, \phi], \phi \in \left\{\frac{2^i\pi}{16}\right\}_{i=0}^{4}$
CFO Δf	$\Delta f \in \{0.1 + 0.2i\}_{i=0}^{4}$
The number of sampling points N	$N = 128$
The number of training samples for GAMC	20000 samples/type/SNR (SNRs = $\{-5, 0, 5\}$ dB)
The number of training samples for FAMC	60000 samples/type/SNR (SNRs = [-5, 5] dB)
The number of test samples N_{test}	30000 samples/type/SNR (SNRs = [-5, 5] dB and $N_{SNR} = 11$)
Maximum training epochs	200
Batch sizes	500
Learning rate η	$\eta = 0.001$

Time (μs/sample) SNR (dB) Method	-5	0	5
FAMC/GAMC (GPU)		37.71	
FAMC/GAMC (CPU)		698.20	
Traditional AMC (CPU)	2346.60	2225.19	1807.83

* Considering that FAMC and GAMC have the same CNN structure, and if ignoring calculation error, they have the same degree of computation complex, i. e., the same unit computation time.

The simulation requires powerful computing resources, so it is conducted on the platform with one Intel i7-8750H CPU and one NVIDIA GTX 1080Ti GPU. The implementation of neural networks relies on Keras 2.2.2 with Tensorflow 1.10 and Python 3.6.5 as the backend. SVM is carried out in Sklearn-Python library. Moreover, Matlab R2018a is applied to build our datasets.

Here, three metrics are applied to evaluate classification performances, and The former two metrics are correct classification probability (CCP) at SNR= i dB: P_{cc}^i and average correct classification probability (AveCCP): $\overline{P_{cc}}$, which are shown as follows.

$$P_{cc}^i = \frac{N_{cc}^i}{N_{test} \times N_{type}} \times 100\%, i \in \{-5, -4, ..., 4, 5\}, \quad (13)$$

$$\overline{P_{cc}} = \frac{\sum_{i=-5}^{5} N_{cc}^i}{N_{test} \times N_{type} \times N_{SNR}} \times 100\%, \quad (14)$$

where N_{cc}^i, N_{test}, and N_{SNR} represent the number of correctly recognized samples at SNR= i dB, the amount of test samples at each type and SNR, and the amount of sampling SNRs, respectively. P_{cc}^i appears with the format of graphs and $\overline{P_{cc}}$ is shown as table format.

For the visualization of the specific classification performance for each modulation type in various AMC methods, the third metric, applied in this paper, is the confusion matrix with dimension 3×3.
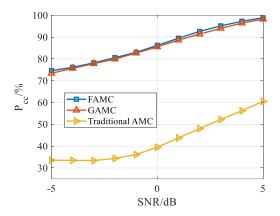

Fig. 6. The classification performance of FAMC, GAMC, and traditional AMC with the condition of CFO and CPO. It can be observed that FAMC and GAMC have perfect and similar CCP at each SNR, and the classification accuracy of FAMC is slightly higher than that of GAMC, but traditional AMC has far weaker performance than these two CNN-based AMC methods.
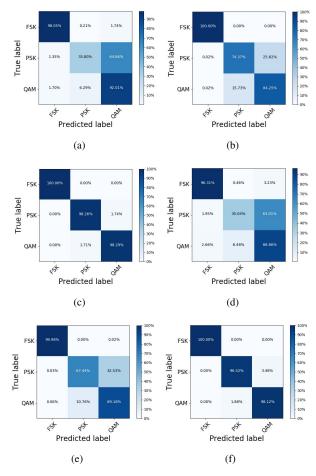

Fig. 7. Confusion matrices of FAMC and GAMC without the consideration of CFO and CPO. FAMC: (a) -5 dB; (b) 0 dB; (c) 5 dB. GAMC: (d) -5 dB; (e) 0 dB; (f) 5 dB.

A. Classification performance and computation complexity of FAMC, GAMC and traditional AMC

The specifies of various AMC methods under WNGN condition and without CFO and CPO are depicted in Fig. 6. From these experimental results, we can observe that the traditional
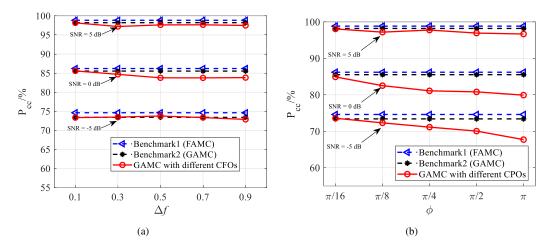
Fig. 9. The classification performance of GAMC considering different CFOs and CPOs. With the increase of the normalized CFO, the classification performance of GAMC is slightly affected. However, the increasing CPO leads to the sharp performance degradation, while it has the limited performance decline at high SNR, such as 5 dB.
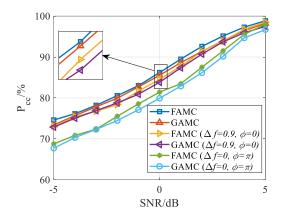

Fig. 8. The specific classification performance of FAMC and GAMC with the considering of CFO ($\Delta f = 0.9$) and CPO ($\phi = \pi$), respectively. FAMC and GAMC with the consideration of CPO have lower P_{cc}^i than that without the consideration of CPO or with the consideration of CFO. In addition, there is the limited performance gap between FAMC and its corresponding GAMC.

TABLE III
THE AVERAGE CORRECT CLASSIFICATION PROBABILITY OF DIFFERENT AMC METHODS.

Methods	$\overline{P_{cc}}$ (%)
FAMC (Benchmark)	86.55
GAMC	85.79
Traditonal AMC	42.85
FAMC ($\Delta f = 0.9, \phi = 0$)	85.69
GAMC ($\Delta f = 0.9, \phi = 0$)	84.86
FAMC ($\Delta f = 0, \phi = \pi$)	82.14
GAMC ($\Delta f = 0, \phi = \pi$)	81.13

AMC method, based on SVM and HOC, has unsatisfactory performances, compared with the CNN-based AMC methods.

P_{cc}^i of GAMC is similar with that of FAMC, and their maximum gap of P_{cc}^i only can reach up to 1.22% at SNR = 2 dB or -5 dB. It means that GAMC and FAMC has few performance gap. In addition, this phenomenon also appears in the other metric: $\overline{P_{cc}}$, and their performance gap of $\overline{P_{cc}}$ is less than 1%, which is shown in Table III.

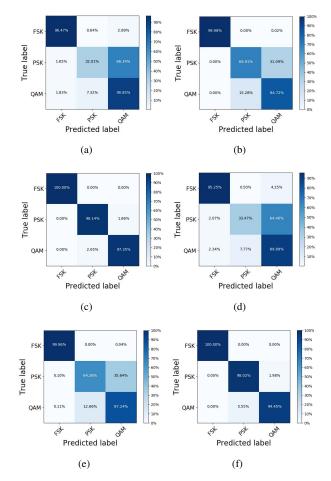The confusion matrices of FAMC and GAMC at three SNRs are given in Fig. 7. Compared with the confusion matrices

Fig. 10. Confusion matrices of FAMC and GAMC with the consideration of CFO and $\Delta f = 0.9$. The confusion matrices of FAMC at SNRs = -5, 0, 5 dB are given in (a), (b), and (c). Corresponding confusion matrices of GAMC are shown in (d), (e), and (f).

of FAMC and GAMC at the same SNR, they are extremely similar. At low SNR, both FAMC and GAMC can barely identify PSK and QAM, but their classification performances are gradually improved, as SNR increases. However, FSK
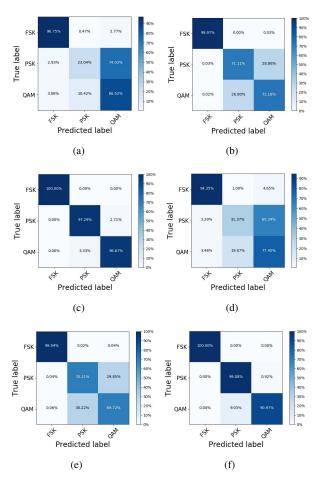
Fig. 11. Confusion matrices of FAMC and GAMC with the consideration of CPO and $\phi = \pi$. (a) FAMC at SNR = -5 dB; (b) FAMC at SNR = 0 dB; (c) FAMC at SNR = 5 dB; (d) GAMC at SNR = -5 dB; (e) GAMC at SNR = 0 dB; (f) GAMC at SNR = 5 dB.

TABLE IV
THE AVERAGE CORRECT CLASSIFICATION PROBABILITY OF GAMC CONSIDERING DIFFERENT CFOs.

Δf	$\overline{P_{cc}}$ (%)
0	85.79
0.1	85.83
0.3	85.21
0.5	85.08
0.7	84.89
0.9	84.86

TABLE V
THE AVERAGE CORRECT CLASSIFICATION PROBABILITY OF GAMC CONSIDERING DIFFERENT CPOs.

ϕ	$\overline{P_{cc}}$ (%)
0	85.79
$\pi/16$	85.45
$\pi/8$	83.99
$\pi/4$	82.64
$\pi/2$	82.10
π	81.13

CFO has few effect on FAMC and GAMC, while their identification capabilities are both worse under the influence of CPO.

What's more there are almost no expansion of the classification performance gap between FAMC and GAMC, because their $\overline{P_{cc}}$ gap of FAMC and GAMC are within or slightly higher than 1%. In addition, their maximum P_{cc}^i gap is 1.01% with the consideration of CFO, while the value just reaches up to 1.49%, when considering CPO. The confusion matrices in Fig. 10 and Fig. 11 also illustrates that FAMC and GAMC have similar performances under any circumstances in this paper.

Then, the classification performances of GAMC under the condition of different CFOs and CPOs is considered, and their experimental results are shown in Fig. 9, Table IV and Table V. As mentioned before, P_{cc}^i and $\overline{P_{cc}}$ of GAMC is almost unaffected by normalized CFO.

As is shown in Fig. 9(b) and Table V, the influence of CPO is weak at $\phi = \frac{\pi}{16}$ or high SNR, such as 5 dB. However, the influence of CPO is gradually getting strong with the increase of the value of ϕ. Phase offset correction algorithms should be considered to aid CNN-based AMC methods for the improvement of classification performances, when ϕ is too large.

C. Generalization capabilities under varying SNR conditions

In the former sections, the specific classification performances of FAMC and GAMC have been introduced carefully through three metrics, and it can be concluded that there are extremely weak performance gaps between FAMC and GAMC. In this section, it is illustrated that GAMC has more powerful generalization capabilities than FAMC at the expense of the slight performance loss. What's more, we give two sets of simulation results for generalization capabilities. The one set of results is tested within the ranges of training SNRs, i. e., SNRs = [-5, 5] dB, and the other set is out of the ranges, including SNRs = [-15, -6] dB and [6, 15] dB.

1) Within the range of training SNRs: To compare the generalization capabilities between FAMC and GAMC under varying noise conditions, we depict three curves of CNN (j dB) in FAMC trained at different SNRs (i. e., j = -5, 0, 5) in Fig. 12, and we tested them at SNRs ranging from -5 dB to 5 dB.

The CNN (j dB) in FAMC performs well when testing SNRs are close to j dB (1 dB error can be allowed), but its performance gets worse at other SNRs, which means that the FAMC does not have higher robustness and generalization

can be precisely distinguished from other modulation types in these two AMC methods, even at low SNR, such as -5 dB.

Besides, computation cost or computation complex is another metric to measure AMC methods. In this paper, unit computation time is applied as a metric for the evaluation of three methods, which are shown in Table II. The time represents the average computation time of single IQ sample after testing a large number of samples, and it is calculated in the same platform listed in the head of this section.

From Table II, it can be observed that FAMC and GAMC not only on GPU but also on CPU have far higher computation speed than traditional AMC. These results demonstrate that the CNN-based AMC methods are more efficient than the traditional AMC, particularly in communication systems equipped with GPU.

B. Classification performance of FAMC and GAMC considering CFO and CPO

In this section, the traditional AMC is not considered, because of the weak classification performance and slow computation speed. The influence of CFO and CPO for the classification performances of the CNN-based AMC method is shown in Fig. 8 and Table III. It is noted that the normalized
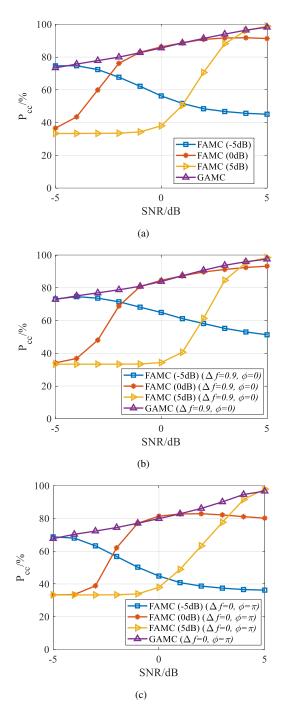
(a)

(b)

(c)

Fig. 12. Generalization capability of FAMC and GAMC. (a) is just under WNGN condition without the consideration of CFO and CPO; (b) and (C) are under WNGN condition considering CFO and CPO, respectively.

It is demonstrated that GAMC can extract more robust features from the mixed datasets than FAMC, and GAMC has more powerful generalization capabilities than FAMC. In addition, we only prepare eleven trained CNN models, i. e., CNN (i dB) and $i \in \{-5, -4, ..., 4, 5\}$. So we have to apply CNN (-5 dB) or CNN (5 dB) into noise conditions with SNRs = [-15, -6] dB or [6, 15] dB.
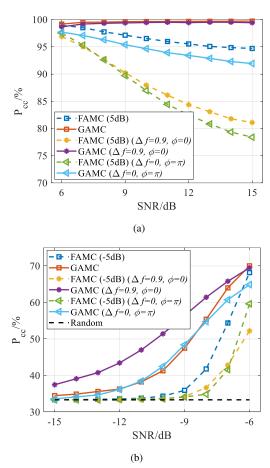

(a)

(b)

Fig. 13. Generalization capabilities of FAMC and GAMC out of the range of training SNRs.

capabilities. On the contrary, Fig. 12 demonstrates that GAMC can work well at all the testing SNRs, whether or not CFO and CPO are considered.

2) Out of the scope of training SNRs: In order to make the future compare the generalization capabilities of FAMC and GAMC, we present the simulation results for SNRs outside the range of training SNRs in Fig. 13. In Fig. 13(a), the CCPs of GAMC at SNRs, which is higher than 5 dB, is far beyond that of FAMC. The similar simulation results also appear in Fig. 13(b), when SNR is lower than -5 dB.

V. CONCLUSION

In this paper, we have proposed a CNN-based GAMC method with better robustness under varying noise conditions. Compared with the traditional AMC method, the classification accuracy of GAMC is far beyond. Besides, GAMC is more robust than FAMC at the cost of negligible performance loss, because the CNN in GAMC is trained by a mixed IQ dataset containing received signals with SNRs of -5 dB, 0 dB and 5 dB, and it can be applied to recognize modulation types of signals with uncertain SNR from -5 dB to 5 dB. Moreover, our proposed GAMC method is more practical than the CNN-based FAMC methods, because we just have to train one CNN model in GAMC rather than many CNN models in FAMC, which means less device memory assumption. In addition, precise SNR estimation is unnecessary for GAMC to choose suitable CNN models. Hence, our proposed CNN-based GAMC method is meaningful for practical applications.

What's more, the CNN in FAMC works perfectly, yet this CNN can also be applied into GAMC with more complex mixed dataset. It is demonstrated by experimental results that the CNN in GAMC has the capability to extract universal and robust features from mixed dataset with different SNRs. It also illustrates that the CNN, designed by human experience here, has massive redundancy and its operation speed is also limited. Thus, our future work will focus on finding more effective and streamlined neural network model, such as network slimming algorithm [40] and neural architecture search (NAS) [41].

REFERENCES

[1] H. Wu, M. Saquib, Z. Yun, "Novel Automatic Modulation Classification Using Cumulant Features for Communications via Multipath Channels," *IEEE Transactions on Wireless Communications*, vol. 7, no. 8, pp. 3098 – 3105, Aug. 2008.

[2] L. Haring, Y. Chen, A. Czylwik, "Automatic Modulation Classification Methods for Wireless OFDM Systems in TDD Mode," *IEEE Transactions on Communications*, vol. 58, no. 9, pp. 2480 – 2485, Sept. 2010.

[3] M. Aslam, Z. Zhu, A. Nandi, "Automatic Modulation Classification Using Combination of Genetic Programming and KNN," *IEEE Transactions on Wireless Communications*, vol. 11, no. 8, pp. 2742 – 2750, 2012.

[4] S. Hakimi, and G. Abed Hodtani, "Optimized Distributed Automatic Modulation Classification in Wireless Sensor Networks Using Information Theoretic Measures," *IEEE Sensors Journal*, vol. 17, no. 10, pp. 3079 – 3091, Oct. 2017.

[5] Z. Zhu, A. Nandi, Automatic Modulation Classification: Principles, Algorithms and Applications, John Wiley & Sons, 2015.

[6] S. Rajendran, W. Meert, D. Giustiniano, V. Lenders, S. Pollin, "Deep Learning Models for Wireless Signal Classification With Distributed Low-Cost Spectrum Sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 3, pp. 433-445, Mar. 2018.

[7] F. Meng, P. Chen, L. Wu, and X. Wang, "Automatic Modulation Classification: A Deep Learning Enabled Approach," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 10760 – 10772, 2018.

[8] F. Hameed, O. A. Dobre, D. C. Popescu, "On the Likelihood-based Approach to Modulation Classification," *IEEE Transactions on Wireless Communications*, vol. 8, no. 12, pp. 5884-5892, 2009.

[9] Z. Yin, et al., "Robust Automatic Modulation Classification Under Varying Noise Conditions," *IEEE Access*, vol. 5, no. 1, pp. 19733 – 19741, 2017.

[10] Z. Fadlullah, et al., "State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2432 – 2455, 2017.

[11] Y. Zhou, Z. Fadlullah, B. Mao, N. Kato, "A Deep-Learning-Based Radio Resource Assignment Technique for 5G Ultra Dense Networks," *IEEE Network*, vol. 32, no. 6, pp. 28-34, Dec. 2018.

[12] B. Mao, et al., "A Novel Non-Supervised Deep Learning Based Network Traffic Control Method for Software Defined Wireless Networks," *IEEE Wireless Communications Magazine*, vol. 25, no. 4, pp. 74 – 81, 2018.

[13] C. K. Wen, W. T. Shih, and S. Jin, "Deep Learning for Massive MIMO CSI Feedback," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 748–751, 2018.

[14] J. Wang, et al., "Device-Free Wireless Localization and Activity Recognition: A Deep Learning Approach," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 7, pp. 44 – 55, Jul. 2017.

[15] Y. Xing, et al., "Driver Activity Recognition for Intelligent Vehicles: A Deep Learning Approach," *IEEE Transactions on Vehicular Technology*, to be published, doi: 10.1109/TVT.2019.2908425

[16] P. Zhou, et al., "Deep Learning-Based Beam Management and Interference Coordination in Dense mmWave Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 592 – 603, Jan. 2019.

[17] H. Jiang, et al., "A 3D Non-Stationary Wideband Geometry-Based Channel Model for MIMO Vehicle-to-Vehicle Communication Systems," *IEEE Transactions on Vehicular Technology*, to be published, 2019.

[18] X. Sun, et al., "ResInNet: A Novel Deep Neural Network with Feature Reuse for Internet of Things," *IEEE Internet Things Journal*, vol. 6, no. 1, pp. 679 – 691, 2018.

[19] H. Huang, Y. Song, and J. Yang, "Deep-Learning-based Millimeter-Wave Massive MIMO for Hybrid Precoding," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 3027 – 3032, 2019.

[20] H. Huang, et al., "Deep Learning for Super-Resolution Channel Estimation and DOA Estimation based Massive MIMO System," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8549 – 8560, 2018.

[21] M. Liu, et al., "DSF-NOMA: UAV-Assisted Emergency Communication Technology in a Heterogeneous Internet of Things," *IEEE Internet Things Journal*, to be Publish doi 10.1109/JIOT.2019.2903165.

[22] F. Tang, Z. M. Fadlullah, B. Mao, and N. Kato, "An Intelligent Traffic Load Prediction Based Adaptive Channel Assignment Algorithm in SDN-IoT: A Deep Learning Approach," *IEEE Internet Things Journal*, vol. 5, no. 6, pp. 5141 – 5154, 2018.

[23] F. Tang, B. Mao, Z. Fadlullah, and N. Kato, "On a Novel Deep-Learning-Based Intelligent Partially Overlapping Channel Assignment in SDN-IoT," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 80 – 86, 2018.

[24] W. Kang and D. Kim, "DeepRT: A Predictable Deep Learning Inference Framework for IoT Devices," *IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pp. 279 – 280, 2018.

[25] H. Huang, et al., "Unsupervised Learning Based Fast Beamforming Design for Downlink MIMO," *IEEE Access*, vol. 7, no. 1, pp. 7599 – 7605, 2019.

[26] H. Huang, et al., "Fast Beamforming Design via Deep Learning," *IEEE Transactions on Vehicular Technology*, to be published, doi: 10.1109/TVT.2019.2949122.

[27] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep Learning for an Effective Nonorthogonal Multiple Access Scheme," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8440 – 8450, 2018.

[28] M. Liu, T. Song, and G. Gui, "Deep Cognitive Perspective: Resource Allocation for NOMA based Heterogeneous IoT with Imperfect SIC," *IEEE Internet Things Journal*, vol. 6, no. 2, pp. 2885 – 2894, Feb. 2019.

[29] M. Liu, et al., "Deep Learning-Inspired Message Passing Algorithm for Efficient Resource Allocation in Cognitive Radio Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 641 – 653, 2018.

[30] T. O'Shea and J. Hoydis, "An Introduction to Deep Learning for the Physical Layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563 – 575, 2017.

[31] B. Tang, Y. Tu, Z. Zhang, and Y. Lin, "Digital Signal Modulation Classification With Data Augmentation Using Generative Adversarial Nets in Cognitive Radio Networks," *IEEE Access*, vol. 6, pp. 15713 – 15722, 2018.

[32] Y. Lin, Y. Tu, Z. Dou, and Z. Wu, "The Application of Deep Learning in Communication Signal Modulation Recognition," in *IEEE/CIC International Conference on Communications in China (ICCC)*, 2017, pp. 1 – 5.

[33] Y. Tu and Y. Lin, "Deep Neural Network Compression Technique Towards Efficient Digital Signal Modulation Recognition in Edge Device," *IEEE Access*, vol. 7, pp. 58113 – 58119, 2019.

[34] Y. Wang, M. Liu, J. Yang, and G. Gui, "Data-Driven Deep Learning for Automatic Modulation Recognition in Cognitive Radios," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 4074 – 4077, 2019.

[35] D. Middleton, "Non-Gaussian Noise Models in Signal Processing for Telecommunications: New Methods an Results for Class A and Class B Noise Models," *IEEE Transactions on Information Theory*, vol. 45, no. 4, pp. 1129 – 1149, 1999.

[36] S. Hu, et al., "Robust Modulation Classification under Uncertain Noise Condition Using Recurrent Neural Network," in *IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1 – 7.

[37] A. Hazza, M. Shoaib, S. A. Alshebeili, and A. Fahad, "An Overview of Feature-Based Methods for Digital Modulation Classification," in *International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, 2013, pp. 1 – 6.

[38] K. Hassan, I. Dayoub, W. Hamouda, et al., "Blind Digital Modulation Identification for Spatially-correlated MIMO Systems," *IEEE Transactions on Wireless Communications*, vol. 11, no. 2, pp. 683 – 693, 2011.

[39] S. Ioffe, C. Szegedy, "Batch Normalization: Accelerating Deep NetworkTraining by Reducing Internal Covariate Shift" arXiv preprint, arXiv:1502.03167, 2015.

[40] Z. Liu, J. Li, Z. Shen, et al, "Learning Efficient Convolutional Networks Through Network Slimming," in *ICCV*, Venice, Italy, Oct. 22–29, 2017, pp. 2736 – 2744.

[41] C. Liu, B. Zoph, M. Neumann, et al, "Progressive Neural Architecture Search," in *European Conference on Computer Vision (ECCV)*, 2018, pp. 19 – 34.