Few-Shot Modulation Classification Method Based on Feature Dimension Reduction and Pseudo-Label Training

YUNHAO SHI[ID], **HUA XU**[ID], **LEI JIANG**[ID], **AND YINGHUI LIU**[ID]

Information and Navigation College, Air Force Engineering University, Xi'an 710077, China

Corresponding author: Yunhao Shi (shiyunhaoai@163.com)

ABSTRACT In modulation classification domain, handcrafted feature based method can fit well from a few labeled samples, while deep learning based method require a large amount of samples to achieve the superior classification performance. In order to improve the modulation classification accuracy under the constraint of limited labeled samples, this paper proposes a few-shot modulation classification method based on feature dimension reduction and pseudo-label training (FDRPLT), which combines handcrafted feature based method with deep learning based method. First, an optimal low-dimensional feature subset is created by the combination of the handcrafted features and autoencoder-extracted features post-processed by a feature selection algorithm. Then, a fully connected network (FCN), trained on a small number of labeled signals, is designed for the automated annotating, where unlabeled samples can be annotated and used for the later convolution neural network (CNN) training. The simulation results show that the classification accuracy of eight kinds of modulation types can reach to 98.3% when the SNR is 20dB.

INDEX TERMS Autoencoder, feature dimension reduction, modulation classification, pseudo-label.

I. INTRODUCTION

Automatic modulation classification (AMC) refers to recognizing the modulation type of target signal automatically. As an important research topic in recent years, AMC has been widely used in many fields, such as communication reconnaissance, blind signal processing, surveillance and threat analysis.

In general, the existing methods of AMC can be divided into two categories: traditional handcrafted feature based methods and deep learning based methods. In recent years, both these two kinds of methods have achieved outstanding performance. In traditional handcrafted feature based methods domain, the authors of [1]–[3] classified modulation signals by using high-order cumulants, and the authors of [4], [5] classified signals by extracting cyclic spectrum features. The authors of [6], [7] used information entropy features for AMC. In deep learning based methods domain, authors summarized the typical AMC methods based on deep learning in recent years in [8]. The authors of [9], [10] started using supervised learning for AMC in 2016 firstly,

they used convolution neural network (CNN) to construct an end-to-end learning model, and successfully identified 11 digital signals with different modulations, including Wide Band Frequency Modulation (WBFM), Double Side Band (DSB), Binary Phase Shift Keying (BPSK) and 16 Quadrature Amplitude Modulation (QAM). In [11], Short time Fourier transform (STFT) is utilized to convert signals from time domain to time-frequency domain, and then CNN is used to extract the time-frequency features. The experiment shows that several kinds of modulation types including 2 Frequency Shift Keying (FSK), 4FSK and 8FSK can be classified with an accuracy rate more than 90% even when the signal to noise ratio (SNR) is low to −4dB. The authors of [12] proposed a joint noise estimation algorithm with a clever network structure, where the original signal and SNR are input into neural network at the same time. The simulation result shows that the classification accuracy under different frequency offset is very close to the performance limit under different SNR. The authors of [13] extracted time-frequency plane features from the Smoothed Pseudo Wigner-Ville Distribution (SPWVD) and Born-Jordan distribution (BJD) transformation of signals by using CNN, and then classified them into 8 modulation types including BPSK, QPSK and

---

The associate editor coordinating the review of this manuscript and approving it for publication was Mouloud Denai[ID].

Orthogonal Frequency Division Multiplexing (OFDM). The experiment indicates that the classification accuracy can still reach 92.5% when the SNR is low to -4dB.

Although the deep learning based methods surpass the traditional handcrafted feature based method in classification performance in recent years, it requires significantly larger number of labeled samples for training. Once the number of labeled samples is insufficient, the classification performance of the deep learning based methods will decline sharply. With the diversification of signal acquisition methods and the rapid development of storage technology, it is simple to obtain a large number of unlabeled signals. However, it is very difficult to obtain the same amount of labeled signals, because the work of data labelling requires a lot of manpower, material resources and time. Besides, there exist a large number of signals in the electromagnetic spectrum in real environment. It is unrealistic to label all these signals one by one after collection due to the time requirement for rapidly changing situations. Therefore, it is particularly important to study the few-shot modulation classification problem with insufficient labeled signals.

Based on a large number of results from expert research, it is found that traditional handcrafted feature based methods do not need a large number of labeled samples, while deep learning based methods need massive labeled samples, that is because handcrafted method reduces the signal to a low-dimensional by processing the signal sequence in a fixed mode while deep learning method generally process time sequence or timefrequency transformation plane having highdimensional data. Generally, the number of samples required by classifier is proportional to the feature dimension. Hence, dimension reduction is a core to solve few-shot problem. If low-dimensional features can be used to represent the original time sequence, the training samples needed by the classifier can be greatly reduced.

Nowadays, the research on few-shot modulation classification is still in its infancy, but some semi-supervised learning algorithms have appeared in other fields. Semi-supervised learning solves the problem of how to improve the model performance when the number of labeled samples is insufficient. The primary principle of semisupervised learning is to use unlabeled data for optimization. Although unlabeled samples have no label information to use, it is obtained from the same source independently and in the same distribution as labeled samples. For this reason, the information they contain is of great benefit to optimize model. Semi-supervised learning methods mainly include self-training learning method [14], generative learning method [15] and semi supervised support vector machine [16].

In order to make full use of unlabeled signals for few-shot modulation classification, this paper proposes a classification scheme based on feature dimension reduction and pseudo-label training (FDRPLT). It mainly consists of three parts: pseudo-label annotator training, pseudolabel labeling and deep neural network training. In the first part, handcrafted features and unsupervised learning features are extracted and selected. Then, a fast converging classifier is trained by a small number of samples with low-dimensional features. In the second part, the classifier is used to generate label for unlabeled samples. In the last part, all real-label samples and pseudo-label samples are sent into deep neural network for training. The experimental results show that our algorithm has an excellent performance under the condition of small samples. Specifically, the classification accuracy of this method can reach 98.3% with the SNR at 20dB.

The rest of this paper is organized as follows. Section II introduces the framework of our algorithm. Section III describes the design of pseudo-label annotator. Section IV shows the labeling process and Section V introduces the deep neural network structure. Section VI shows the simulation result and the last section VII concludes this paper.

II. FRAMEWORK OF FDRPLT

The mainly contribution of FDRPLT is find a connection with traditional handcrafted feature based method and deep learning based method. The key idea of FDRPLT is to design a pseudolabel annotator to provide sufficient reliable labeled samples for deep learning based methods. As shown in Fig. 1, the algorithm is divided into three parts. The first part is the generation of pseudo-label annotator. The traditional method and unsupervised autoencoder are used to extract features of original time sequences. After merging these two types of features, a feature selection algorithm is used to generate the optimal feature subset. Then, a fast converging classifier is trained by a small number of low-dimensional labeled signals. The second part is pseudo-label generation, which uses the pseudo-label annotator trained in the first part to label the unlabeled samples. Then, samples with pseudo-label are combined with samples having real-label to optimize the classifier until meet the iteration conditions. The third part is deep neural network training, where all real-label samples and pseudolabel samples are combined to train a deep neural network, and the trained network is used to predict test signals.

The core of this algorithm is the generation of pseudo-label annotator, only when the classification performance of annotator is as high as possible can the performance of deep neural network be guaranteed. As long as the performance of annotator is excellent, it can provide deep neural network with sufficient and reliable training samples.

III. GENERATION OF PSEUDO-LABEL ANNOTATOR

Pseudo-label annotator is to improve the signal classification accuracy to a higher level as much as possible when there is only a small number of labeled samples, so as to ensure the reliability of pseudo-label samples provided for deep neural network. Because samples required by classifier increase exponentially with the feature dimension, this paper resolves the constrain imposed by limited number of samples using self-designed pseudo-label annotator and boosts the classification performance using the optimized feature set.
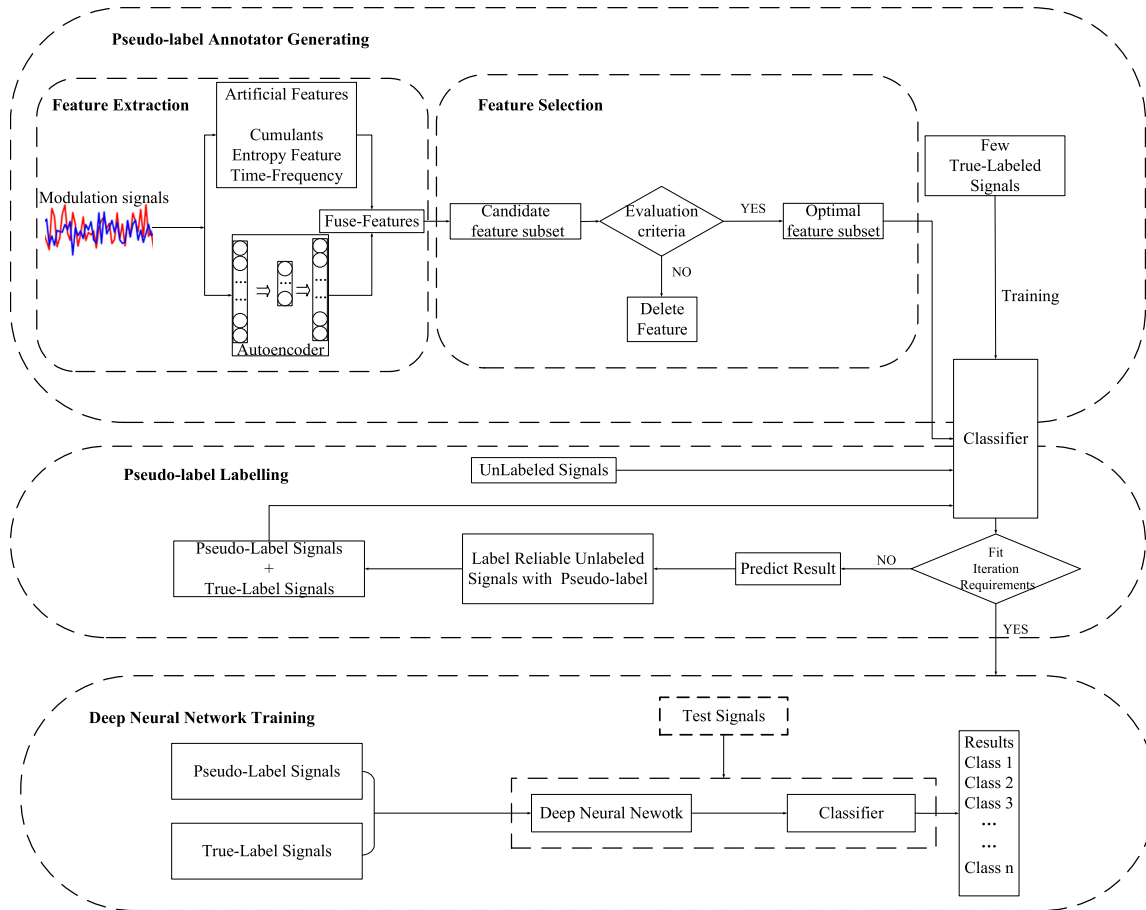
IEEE Access

FIGURE 1. Algorithm block diagram.

A. FEATURE EXTRACTION

1) HANDCRAFTED FEATURES

In AMC domain, many researchers have done a lot of feature engineering and designed many effective features for modulation classification. Inspired by previous research result, in this paper, we extract three kinds of features with strong representativeness and high discrimination as signal handcrafted features, including highorder cumulants, information entropy features and time-frequency features.

a: HIGH-ORDER CUMULANTS FEATURES

In order to extract high-order cumulants, high-order moment of signal should first be calculated. The high-order moment of $x(n)$ can be calculated by the following equation:

$$M_{pq} = E[x(n)^{p-q}(x^*(n))^q] \tag{1}$$

In this paper, the following high-order cumulants are selected through [17], [18], which have been proved to have good discrimination performance in modulation classification:

$$C_{40} = M_{40} - 3M_{20}^2 \tag{2}$$

$$C_{42} = M_{42} - M_{20}^2 - 2M_{21}^2 \tag{3}$$

$$C_{60} = M_{60} - 15M_{40}M_{20} + 30M_{20}^3 \tag{4}$$

$$C_{61} = M_{61} - 5M_{40}M_{21} - 10M_{20}M_{41} + 30M_{21}M_{20}^3 \tag{5}$$

$$C_{63} = M_{63} - 6M_{41}M_{20} - 9M_{21}M_{42} + 18M_{21}M_{20}^2 + 12M_{21}^3 \tag{6}$$

b: ENTROPY FEATURES

Entropy is an index that used to evaluate the mean uncertainty of signal or system. This paper intends to extract power spectrum entropy, singular entropy and energy spectrum entropy of signal as features [19], [6]. Assuming that the length of time sequence X is L, the discrete Fourier transform of sequence X is:

$$y(k) = \sum_{n=0}^{L-1} x(n)e^{-i\frac{2\pi}{N}nk} = \sum_{n=0}^{L-1} x(n)W_N^{nk} \tag{7}$$

where $W = e^{-i\frac{2\pi}{N}}$, k represents the k-th spectrum of Fourier transform, N represents the number of transformed points. Generally, N is the integer power of 2 and close to the length of sequence X. Power spectrum of the spectrum sequence Y at $y(k)$ is calculated:

$$S(k) = \frac{1}{N} |y(k)|^2, \quad k = 0, 1, 2 \ldots N - 1 \tag{8}$$

Denote

$$p_k = \frac{S(k)}{\sum\limits_{k=0}^{N-1} S(k)} \qquad (9)$$

By substituting (9) into the Shannon entropy calculation equation, the power spectrum Shannon entropy can be obtained. For random variables X, the equation of Shannon entropy is shown as follows,

$$H(X) = \sum_{i=1}^{m} p(x_i) \log \frac{1}{p(x_i)} \qquad (10)$$

where H represents the entropy value, p_i represents the signal probability distribution.

In recent years, singular spectrum analysis is a very popular method to study nonlinear time sequence. Suppose a discrete time sequence is:

$$X = [x_1, x_2, x_3 \ldots x_N] $$

Firstly, the sequence is segmented with length m. The phase space matrix is reconstructed as follows:

$$M = \begin{bmatrix} x_1 & x_{1+n} & \cdots & x_{1+(m-1)n} \\ x_2 & x_{2+n} & \cdots & x_{2+(m-1)n} \\ \cdots & \cdots & \cdots & \cdots \\ x_{N-(m-1)n} & x_{N-(m-1)n+n} & \cdots & x_N \end{bmatrix}_{K*J} \qquad (11)$$

The singular value decomposition of (11) can be obtained as:

$$M_{K*J} = U_{K*K} \Sigma_{K*J} V_{J*J}^T \qquad (12)$$

where U and V are orthogonal matrices, U is the left singular matrix, and V is the right singular matrix. Σ is a diagonal matrix:

$$\Sigma = \begin{bmatrix} \sigma_1 & & & \cdots & \\ & \sigma_2 & & \cdots & \\ & \cdots & \sigma_k & \cdots & \\ & & & \cdots & \end{bmatrix}_{K*J}$$

where σ_k represents the singular value of matrix M, and all values in Σ except the diagonal elements are zero. The non-zero elements on the diagonal form the singular value spectrum of the sequence, given by:

$$\sigma = \{\sigma_1, \sigma_2, \ldots \sigma_i, \ldots \sigma_j \,|\, j < K \}$$

p_i is denoted as the ratio of non-zero singular value σ_i to the sum of all non-zero singular values:

$$p_i = \frac{\sigma_i}{\sum\limits_{i=1}^{r} \sigma_i} \qquad (13)$$

The singular value Shannon entropy can be calculated by substituting (13) into (10).

The power spectrum exponential entropy can be obtained by substituting (12) into the equation of exponential entropy. The equation of exponential entropy is shown in (14):

$$H = E[e^{1-p_i}] = \sum p_i e^{1-p_i} \qquad (14)$$

For the sequence X, its energy spectrum is defined as follows:

$$S(\omega) = \frac{1}{N} |X(\omega)|^2 \qquad (15)$$

where $X(\omega)$ represents the Fourier transform of X. Denote p_i as:

$$p_i = \frac{S(i)}{\sum\limits_{i=1}^{N} S(i)} \qquad (16)$$

The energy spectrum exponential entropy can be obtained by taking (16) into the (14).

c: TIME-FREQUENCY FEATURES

The Maximum value of the power density of the normalized-centered instantaneous amplitude can reflect the spectral characteristics of different signals [20], which is defined as follow:

$$\gamma_{\max} = \frac{\max |FFT |a_{cn}(i)||^2}{N_s} \qquad (17)$$

$$a_{cn}(i) = a_n(i) - 1 \qquad (18)$$

$$a_n(i) = a(i)/m_a \qquad (19)$$

$$m_a = \frac{1}{N_s} \sum_{i=1}^{N_s} a(i) \qquad (20)$$

where N_s represents the length of the signal sequence and m_a represents the mean value of the instantaneous amplitude.

2) UNSUPERVISED LEARNING FEATURES

Unsupervised feature extraction method can extract features without using label information, which is helpful for the small sample problem. Autoencoder is a kind of unsupervised learning algorithm which can learn the sparse representation of data. Many scholars have applied it to modulation classification in recent years. For example, the authors of [21] used two parallel autoencoders to complete modulation classification, and the authors of [22] used convolution autoencoder to directly act on time sequence to extract features.

Generally, the structure of an autoencoder includes an encoder and a decoder. The encoder projects the input samples into the hidden layer with a lower dimension and the decoder converts it back to the original feature space and aims to faithfully reconstruct the input data [23]. The function for the autoencoder is to find a low dimensional feature representation of the original high dimensional, complex input.

The autoencoder's structure of our algorithm is shown in Fig. 2, it mainly composed of convolution layers and fully connected layers. In this paper, we use fully connected layers to reduce and enhance the feature dimension. Before feeding the input into the autoencoder, we add noise to the signals intentionally. Since the original non-noise signal need to be reconstructed after the signal passed through the autoencoder,
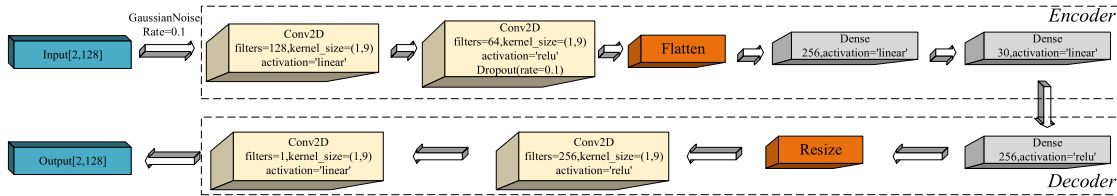
FIGURE 2. Autoencoder structure.

the features extracted by autoencoder have the anti-noise ability and are more robust.

Assuming that the original input signal and the noise-added input signal for the autoencoder network is x and x^*, the loss function of autoencoder can be simplified as,

$$J(x, x^*) = \sum \left\| h(f(x^*)) - x \right\|_2^2 \tag{21}$$

where $f(x)$ represents the encoder function, $h(x)$ represents the decoder function. By minimizing the loss function, the middle layer features can serve as the low-dimensional representation of the high-dimensional signals.

B. FEATURE SELECTION

Different from feature extraction method described in the previous section, feature selection method refers to selecting the most relevant feature subset from the original feature sets. Compared with feature extraction method, feature selection method focuses more on revealing the causal relationship between features and categories. In modulation classification domain, whether to use traditional handcrafted feature or deep learning auto-extracted feature has been controversial, because both of these two methods have their own advantages. Thus, a better approach would be to first combine these two types of features together and drop irrelevant or redundant features using feature selection algorithm afterwards. A good feature selection algorithm can not only help us reduce the number of samples required for the classifier, eliminate redundant or irrelevant features, but also improve the running speed of model, accelerate the convergence speed of algorithm and reduce hardware requirements.

Given the condition of the limited labeled samples and vast unlabeled samples, a semi-supervised feature selection algorithm is considered for the feature selection purposes. In recent years, scholars have proposed a variety of feature selection algorithms, such as [24]–[30], among which the authors of [29] proposed information entropy theory to design the symmetric uncertainty of random variables, which is used to measure the redundancy between features and the correlation between features and categories, and it had good performance in many datasets. The authors of [30] made the most use of unlabeled samples for feature selection.

In this paper, a semi-supervised feature selection algorithm is designed based on the Fast Correlation-based Filter solution (FCBF) [29] and Semi-supervised Representative
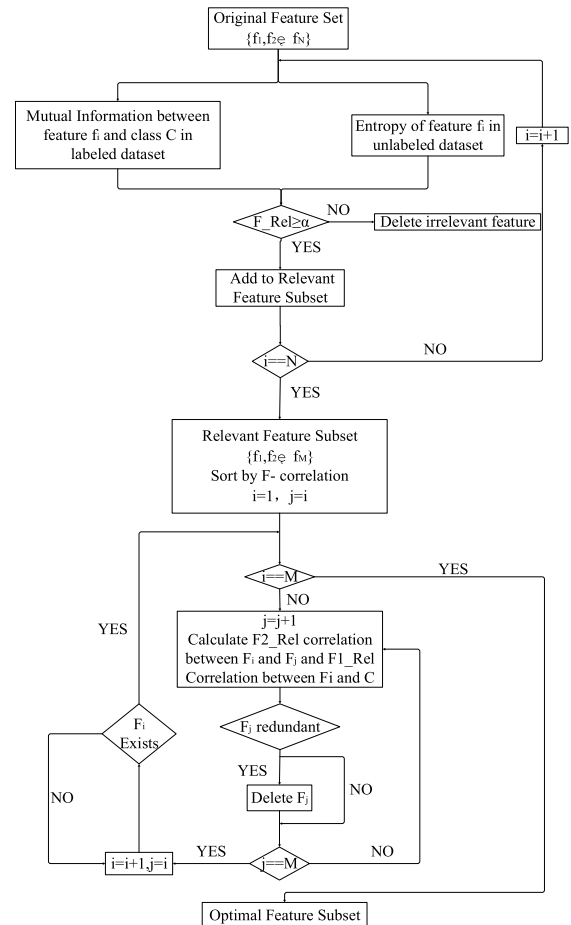

FIGURE 3. Feature selection algorithm.

Feature Selection (SRFS) [30]. The flow chart of this algorithm is shown in Fig. 3, the feature selection process is mainly divided into two steps.

The details of feature selection algorithm are shown as follows:

1) DELETE IRRELEVANT FEATURES

Labeled signals can use label information to directly calculate the mutual information between feature and label as a standard to measure the importance of feature. Suppose the feature is random variable X and the signal label is C; $p(x)$, $p(c)$ and $p(x, c)$ represent the probability density functions of X, C, (X, C) respectively. Then the mutual information of

random variables X and C is defined as:

$$I(X, C) = \sum_{i=1}^{m} \sum_{j=1}^{n} p(x_i, c_j) \log_2 \frac{p(x_i, c_j)}{p(x_i)p(c_j)} \quad (22)$$

Although no label information can be used, self-information contained in the unlabeled signal can also guide feature selection. In information theory, entropy can be used to measure information contained in themselves. The calculation of entropy is shown in equation (10).

After mutual information and selfinformation of each feature are calculated, the F_Rel correlation is used to determine whether feature F_i is irrelevant. The equation of F_Rel is shown as follows,

$$F\_Rel(F_i, C) = \beta I(F_i, C) + (1 - \beta)H(F_i) \quad (23)$$

where β is greater than 0 and less than 1, which can be used to adjust the proportion of mutual information and entropy. In this paper we set $\beta = \sqrt[4]{\frac{N}{N+M}}$, where N represents the number of labeled signals and M represents the number of unlabeled signals.

If F_Rel is greater than the threshold value, it means that the feature is related to categories and carries a large amount of information, then F_i will be added to the relevant feature subset; if F_Rel is less than the threshold value, F_i will be deemed as irrelevant feature to be deleted. This paper designs the relevance threshold value $\alpha = \left\lceil \frac{D}{2\log_2 D} \right\rceil$, where D represents the number of features.

2) DELETE REDUNDANT FEATURES

In the first step, $\{F_1, F_2 \ldots \ldots F_M\}$ was obtained after the irrelevant features were deleted, and the features to be selected were arranged in descending order according to the F_Rel correlation. The higher the value of F_Rel, the higher the rank would be. Then, $F1\_Rel$ correlation between features and categories, as well as $F2\_Rel$ correlation between features and features, are calculated from front to back, the equation of $F1\_Rel$ is shown as follow,

$$F1\_Rel(F_i, C) = \beta UI(F_i)/(H_i) + (1 - \beta)SU(F_i, C) \quad (24)$$

where $UI(F_i)$ represents the mean mutual information value of feature F_i and all other features, and its equation is as follows:

$$UI(F_i) = \frac{1}{n}(H(F_i) + \sum_{j=1:n, j \neq i} I(F_i; F_j))$$

$$= \frac{1}{n}[H(F_i) + \sum_{j=1:n, j \neq i} H(F_i, F_j) - \sum_{j=1:n, j \neq i} H(F_j)] \quad (25)$$

where $SU(F_i, F_j)$ is the symmetric uncertainty of features F_i and F_j, and its calculation formula is shown as follow,

$$SU(F_i, F_j) = 2[\frac{I(F_i; F_j)}{H(F_i) + H(F_j)}] \quad (26)$$

The calculation formula of $F2\_Rel$ is as follow,

$$F2\_Rel(F_i, F_j) = \beta SU(F_i, F_j) + (1 - \beta)USU(F_i, F_j) \quad (27)$$

where $USU(F_i, F_j)$ represents the unsupervised symmetric uncertainty of features F_i and F_j, and its calculation equation is as follows,

$$USU(F_i, F_j) = 2[\frac{UI(F_i; F_j)}{H(F_i) + H(F_j)}] \quad (28)$$

$$UI(F_i; F_j) = UI(F_i) - UI(F_i | F_j)$$
$$= UI(F_i) - \frac{UI(F_i)}{H(F_i)}H(F_i, F_j) + UI(F_i) \quad (29)$$

In this paper, we set (30), (31) as the redundant discriminant condition.

$$F1\_Rel(F_i, C) \geq F1\_Rel(F_j, C) \quad (30)$$
$$F2\_Rel(F_i, F_j) \geq F1\_Rel(F_j, C) \quad (31)$$

If these conditions are met, F_j will be regarded as redundant feature for deletion. In the process of deletion, features ranked first in F_Rel will be retained first, and the final remaining features after iteration consist the optimal feature subset.

C. ANNOTATOR TRAINING

After feature extraction and selection, a shallow fully connected network (FCN), used as a pseudo-label annotator, is trained on a small number of labeled samples with the pre-extracted low dimensional features. The reason why FCN is chosen as the annotator is that FCN not only has strong learning ability, but also has small parameters and is easy to converge. The structure of FCN is shown in Fig. 4. The number of neurons in the input layer is as same as the feature dimension. After input a small number of labeled samples with optimal feature subset, batch normalization (BN) layer is used to normalize features firstly. Because the features sent into FCN are composed of traditional handcrafted features and autoencoder-extracted features, it is necessary to preprocess the features. BN layer is to forcibly impose the distribution of input value of any neuron back to the standard normal distribution with mean value of 0 and variance of 1 by certain normalization means. In this way, the input value falls into the interval where the nonlinear function is more sensitive to the input, that is to say, it can make the gradient larger and avoid the problem of gradient disappearing. Moreover, the larger gradient means that the learning convergence speed is fast and the training speed is greatly accelerated. The output of the BN layer is fed into fully connected layer, with the number of neurons in the fully connected layer being 16,32 and 16 respectively. The output of the fully connected layer is fed into SOFTMAX layer. All the neuron layers use ReLU as the activation function. In addition, in order to improve the network generalization ability, Dropout layer is used to interfere with training after the second and third fully connected layers to prevent network overfitting and improve network generalization ability in
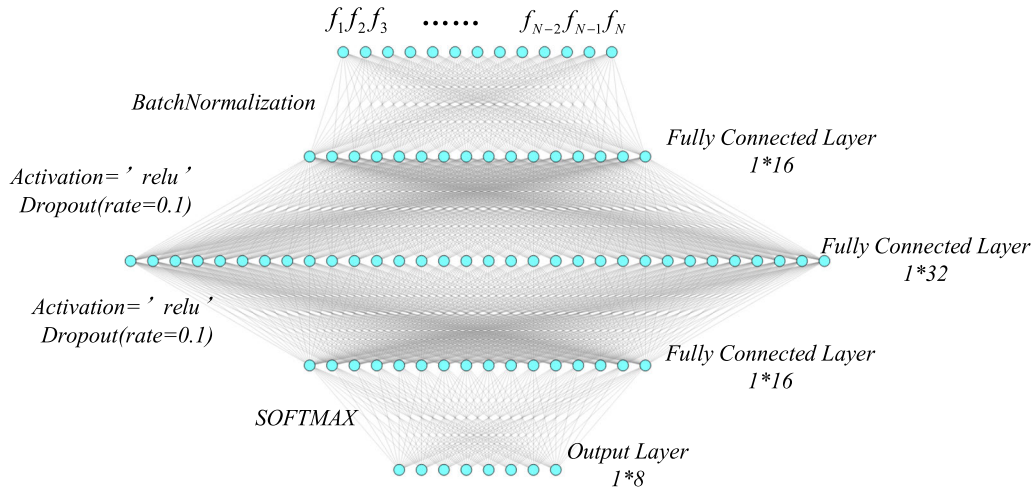
FIGURE 4. FCN structure.

test set. In this paper, Dropout ratio is set to 0.1.

$$L_{total} = \frac{1}{N} \sum_{m=1}^{N} \sum_{i=1}^{C} L_{Real\_Label}(y_i^m, f_i^m)$$
$$+ \mu \frac{1}{N'} \sum_{m=1}^{N'} \sum_{i=1}^{C} L_{Pseudo\_Label}(y_i^m, f_i^m) \quad (32)$$

IV. PSEUDO-LABEL LABELING

After training the pseudo-label annotator, it is used to predict all unlabeled samples. Then all real-labeled samples and pseudo-labeled samples are combined to optimize the annotator until the iterate condition is met. The loss function of pseudo-label annotator is shown in equation (32).

Where $L_{Real\_Label}(y_i^m, f_i^m)$ represents the loss resulting from the samples have real-label, $L_{Pseudo\_Label}(y_i^m, f_i^m)$ represents the loss resulting from the pseudo-labeled samples. N represents the number of real-labeled samples, N' represents the number of pseudo-labeled samples. μ is used to control the proportion of these two types of loss. In general, μ is greater than 0 and less than 1.

In the labeling process, sometimes it is inevitable to label unlabeled samples with false label. Therefore, the reliability of pseudo-label has a decisive impact on the final classification performance. Since the output of SOFTMAX layer is the prediction probability of each category, so this paper designs a reliable condition as follows:

$$p_2 + p_3 \leq p_1 \quad (33)$$

where p_1 represents the maximum probability of the output of SOFTMAX layer, and p_2, p_3 represents the second and third probability of the output of SOFTMAX layer respectively. The labeling criteria is defined such that samples will be labeled only when the maximum probability of output of SOFTMAX is greater than the sum of the second probability and the third probability. Through this condition based on output probability, the reliability of pseudo-labels can be guaranteed to a certain extent.

V. CNN TRAINING

After reliable unlabeled samples are labeled with pseudo-label, all real-label samples and pseudo-label samples are combined and sent to deep neural network for training. Self-designed CNN is selected as the deep neural network in this paper. Different from traditional neural network, CNN can extract features by convolution layer automatically. The structure of CNN is shown in Fig. 5. In order to take full advantage of the feature extraction ability of CNN, IQ sequence is input into CNN directly. The structure of CNN consists of three convolution layers and three fully connected layers. The number of filters of the three convolution layers is 256, 128 and 64 respectively, the filter size of these three convolution layers is 1×9. The convolution layers complete the feature extraction function. Then the extracted features are transformed into a onedimensional sequence and sent into the fully connected layer. The dimension of the three fully connected layers are 256, 128 and 64 respectively. The third fully connected layer sends the features to the SOFTMAX layer for classification, outputs the prediction results. Finally, all the network parameters are optimized through back propagation algorithm. Besides, in order to prevent the network overfitting, Dropout is used after the first and second fully connected layer, the Dropout ratio is set to 0.2. All neurons use ReLU as the activation function.

VI. SIMULATION RESULT

In our simulation, BPSK, 4PSK, 8PSK, 8QAM, 16QAM, 64QAM, 4PAM and 8PAM modulation signals are considered to test the classification performance. The length of samples, L, is equal to 128. The data format of signal is 2×128, including Inphase channel and Quadrature channel. The training set includes 10000 samples for each type of signals and the random SNR is from -10dB to 20dB with an interval of 2dB. There are totally 80,000 samples, including 800 labeled samples and each type has 100 samples, the rest are unlabeled samples. The test set generates 100 samples for
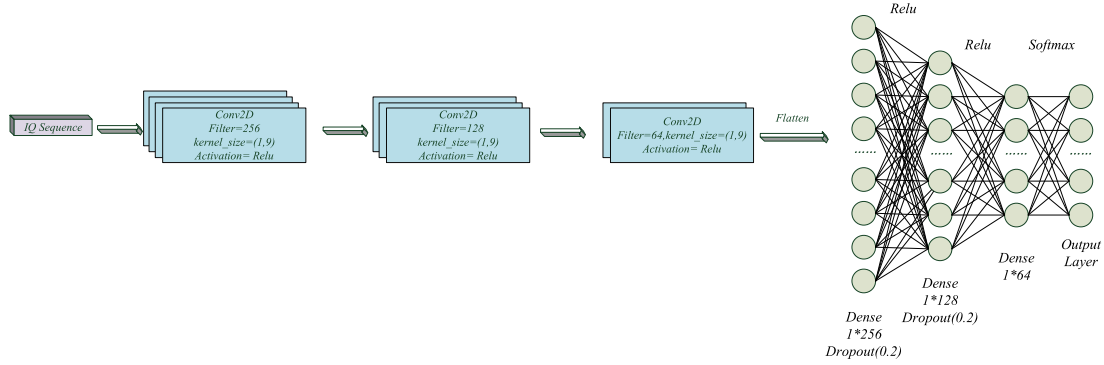
FIGURE 5. CNN structure.

(a)Cumulant C40 (b)Cumulant C63 (c)maximum power density of the normalized center instantaneous amplitude

(d)energy spectrum shannon entropy (e)power spectrum shannon entropy (f)singular spectrum exponential entropy
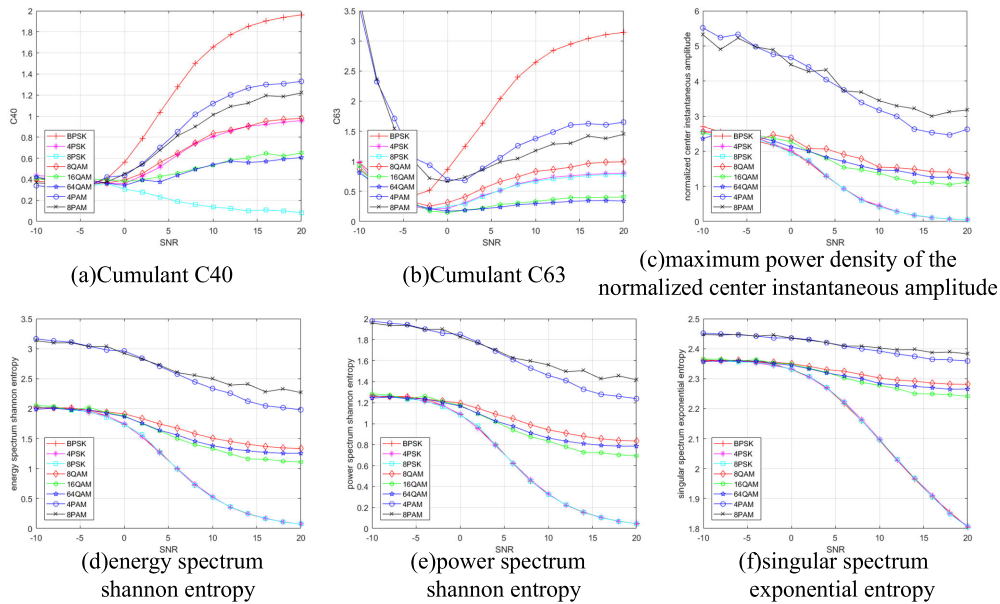
FIGURE 6. Signal feature curve.

each SNR point, ranging from −10dB to 20dB and an interval of 2dB. There are 16 SNR points and 12,800 signals totally. All signals are generated by MATLABR2016a simulation.

The network training is based on TENSORFLOW and KERAS deep learning framework in python. The hardware for calculation support is Intel(R) Core (TM) i7-8700 CPU, and the GPU is NVIDIA GeForce 1060Ti.

A. HANDCRAFTED-FEATURE EXTRACTION

In this section, the performance of handcrafted features proposed in the section III is evaluated. In the experiment, 100 samples are taken from each SNR point between −10dB and 20dB for eight modulation types. Ten kinds of handcrafted features are extracted and averaged. Representative features were plotted as shown in Fig. 4. Besides, in the analysis of signal singular spectrum, because the length of signal L is equal to 128, and the base sequence is randomly generated without periodicity, so we set the segment length m

equal to 43. In the calculation of discrete Fourier transform, the number of Fourier transform points is required to be close to the sequence length, so we set the number of Fourier transform points N equal to 128.

We can see from Fig. 6 that with the continuous rise of SNR, the difference in the eigenvalues of different modulation types gradually increase and tend to be stable, which is helpful for classification. However, at the same time, it can be seen that each feature has an indistinguishable modulation type, so all features are integrated and then selected automatically by the feature selection algorithm in this paper. It can be seen from the experiment that the handcrafted features selected in this paper have good discrimination ability.

B. AUTOENCODER FEATURE EXTRACTION

In this section, ability of extracting features of the autoencoder is tested by experiments. 10000 unlabeled signal
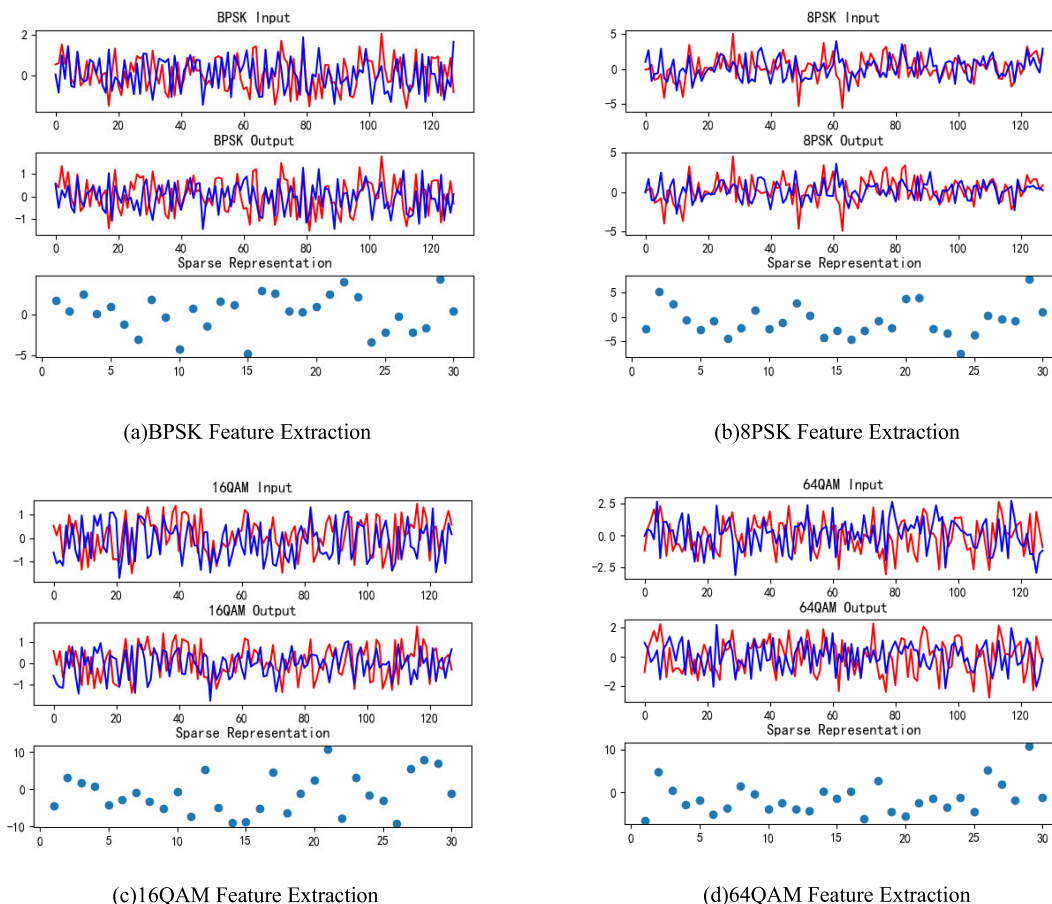
(a)BPSK Feature Extraction

(b)8PSK Feature Extraction

(c)16QAM Feature Extraction

(d)64QAM Feature Extraction

FIGURE 7. Input, output and sparse representation of autoencoder.

samples of each type, totally 80,000 unlabeled samples will be used for unsupervised training. The structure of the autoencoder is shown in section III. Batchsize in the network training is set to 500, and 100 epochs for iterating. MSE is used as the loss function and Adam is used for optimization. The data format of network input and output is 2×128. After training, network parameters are saved, then all signals are input into the network for calculation. Signals are compressed by the encoder layer, and then the output of encoder and decoder are visualized. The results of signal compression and reconstruction are shown in Fig. 7, where shows what the 2×128 input vector looks like, what the 2×128 output representation looks like, and what the $1*30$ sparse representation looks like.

It can be seen from Fig. 7 that the waveform of signals have not changed significantly after the reconstruction of the autoencoder, which means that the loss between input and output is small after the signal reconstruction through the autoencoder, and the low-dimensional $1*30$ features of the middle layer can represent the original IQ data to a certain extent. The training history of the autoencoder is shown in Fig. 8. We can see that the training loss, continues to decrease through the training process, indicating that the
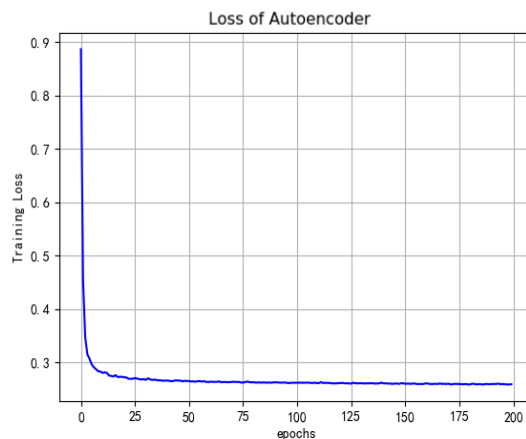

FIGURE 8. Loss of autoencoder.

mapping between input and output signal is successfully established.

In order to verify the performance of the autoencoder designed in this paper, this section compares the autoencoder proposed in [21], [22] and [32]. The number of output neurons of the hidden layer are all set to 30, and the loss curve of each autoencoder is shown in Fig. 9.
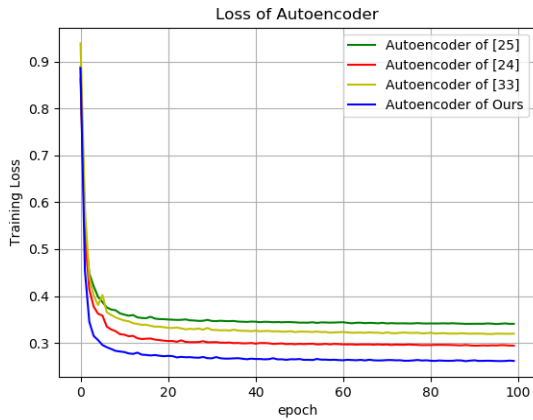
FIGURE 9. Comparison between different autoencoder.

As we can see from the comparison experiment in Fig.9 that the autoencoder in this paper has better reconstruction ability than other autoencoders. From another point of view, the feature extraction ability of our autoencoder is also the strongest.

C. PSEUDO-LABEL ANNOTATOR TRAINING

In this section, we trained a pseudo-label annotator. Firstly, using the autoencoder trained in section B to extract low-dimensional features, then merging all low-dimensional features with handcrafted features extracted in section A, totally 40-dimensional features into the feature selection algorithm. Finally, generating the optimal feature subset and using a small number of labeled signals to train the FCN.

In this experiment, the merged feature set sent to the feature selector is $\{f_1, f_2, f_3 \ldots \ldots f_{39}, f_{40}\}$, among which the first ten-dimensional features $\{f_1 \ldots \ldots, f_{10}\}$ are handcrafted features, representing {Singular spectrum Shannon entropy, Power spectrum Shannon entropy, C_{40}, C_{42}, C_{60}, C_{61}, C_{63}, Energy spectrum Shannon entropy, Singular spectrum Exponential entropy, The Maximum value of the power density of the normalizedcentered instantaneous amplitude}, and the last thirty-dimensional features are the hidden layer features extracted by the autoencoder.

During feature selection, we set paraments $\beta = 0.31, \alpha = 4$, the optimal subset is $\{f_3, f_7, f_8, f_9, f_{10}, f_{12}, f_{16}, f_{19}, f_{21}, f_{27}, f_{32}, f_{36}, f_{37}\}$, 13 features in total, and the feature proportion of selected features is 32.5%.

After feature selection, 800 labeled signals with 13-dimensional features were sent into the FCN for training. After training, the classification accuracy of the FCN is shown in Fig. 10, in which Fig. 10(a) shows the classification confusion matrix when the SNR is 20dB. The classification accuracy of each signal under different SNR is shown in Fig. 10(b). It can be seen that under the condition of high SNR, except 64QAM and 8PAM, the other signals can be accurately recognized.

When the feature selection algorithm is not used, the FCN is trained by 800 labeled samples with different feature

subset. The confusion matrix at 20dB is shown in the Fig. 11, where Fig. 11(a) represents the confusion matrix trained by 800 samples with 10-dimensional handcrafted features, Fig. 11(b) represents the confusion matrix trained by 800 samples with 30-dimensional autoencoder-extracted features, and Fig. 11(c) represents the confusion matrix trained by 800 samples with 40-dimensional merged features.

It can be seen from Fig. 11 that the classification performance of these three methods above are inferior to the algorithm in this paper, whether a single method is adopted or the features are not selected after merging. The highest classification accuracy is 93.1% at 20dB when only used 10-dimensional handcrafted features, and 89.8% at 20dB when only used 30-dimensional autoencoder-extracted features. The highest classification accuracy is 83.7% at 20dB when used 40-dimensional fusion features. The classification accuracy of this paper can reach more than 90% when SNR is greater than 14dB, and the highest classification accuracy can reach 95.8% when SNR is 20dB. It can be seen that the signal classification accuracy is improved after feature fusion and selection. Besides, when the feature dimension is high, the performance of signal classification decreases, for example, when the labeled samples with 40-dimensional features are trained directly, the highest classification accuracy of signals is only 83.7%, that is because the high-dimensional features cannot be fitted well by classifier under the condition of small samples. The classification accuracy of these four kinds of methods at each SNR point is shown in Fig. 12.

In addition, this section selects three kinds of feature selection algorithms for comparison to verify the effectiveness of our feature selection algorithm, namely, mRMR [31], FCBF and SRFS. mRMR and FCBF are supervised feature selection algorithms just using label information, SRFS is semi-supervised feature selection algorithm based on graph theory. Considering the above three algorithms need to set the correlation threshold α and control variable β, we set the correlation threshold $\alpha = 4, \beta = 0.38$, which are as same as the algorithm in this paper. Since it is necessary to evaluate the advantages and disadvantages of the feature subset through classification algorithm and these three feature selection methods compared are independent of classification algorithm. 800 labeled signals with different feature subset selected by each algorithm are sent into the FCN for training and observe its classification accuracy, so as to compare the performance of different feature selection algorithm.

The number of selected features, feature subset and selection time of each algorithm are shown in Table 1. It can be seen that the selecting time of mRMR and FCBF is very short, but it has a large feature subset. Since SRFS algorithm needs to build a directed acyclic graph and select only one representative feature from each sub graph, its subset is concise but time-consuming.

The proportion of selected features of each algorithm is shown in Fig. 13, in which the feature proportion selected
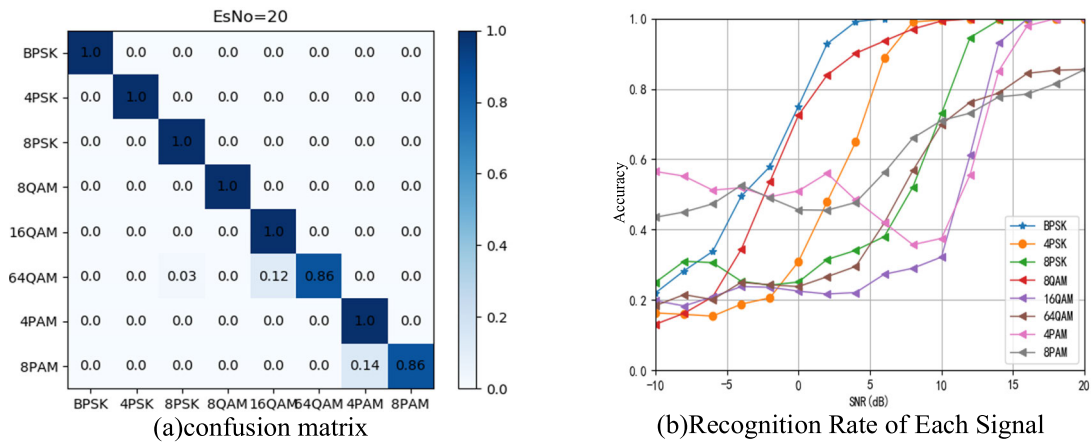
(a)confusion matrix

(b)Recognition Rate of Each Signal

FIGURE 10. Classification performance at 20dB.

(a)10-d Handcrafted Features

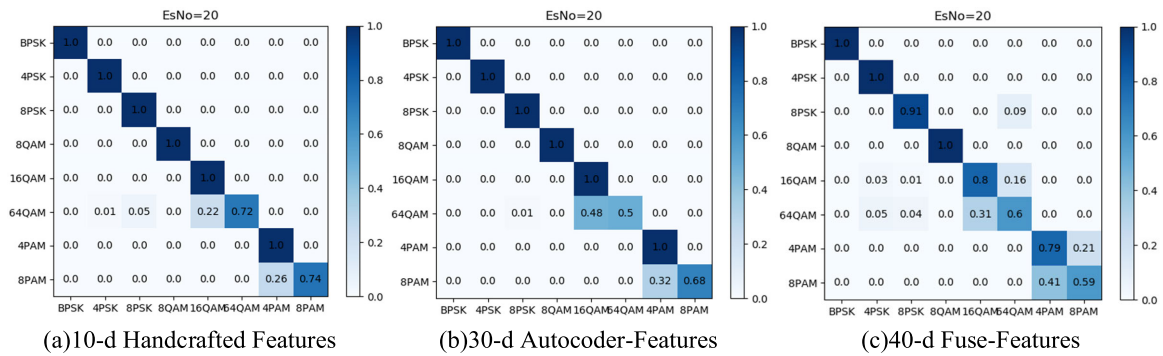(b)30-d Autocoder-Features

(c)40-d Fuse-Features

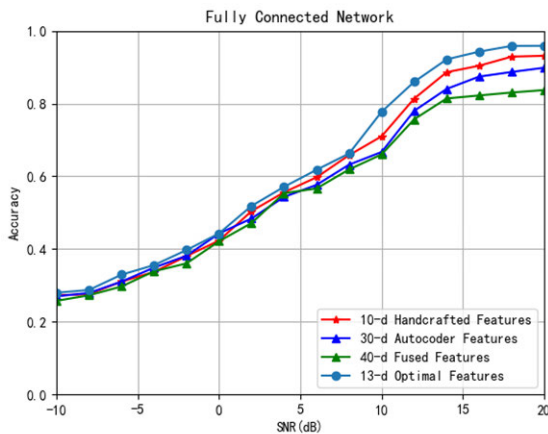**FIGURE 11.** Comparison experiment.

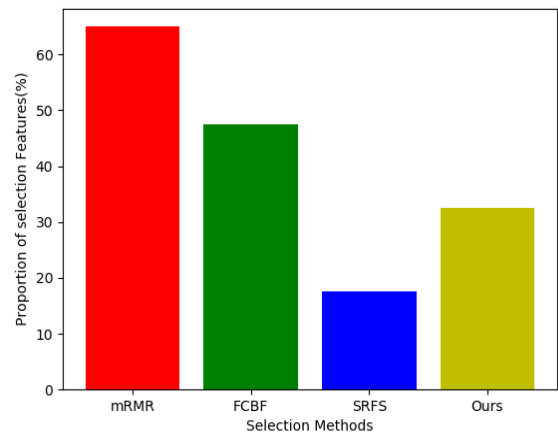**FIGURE 12.** Classification rate of different methods.

FIGURE 13. Proportion of selected features.

by mRMR is 65%, that selected by FCBF is 47.5%, that selected by SRFS is 17.5%, and that selected by this paper is 37.5%.

A small number of labeled samples with different feature subsets selected by each algorithm are sent to FCN for training. The classification accuracy curves under different SNR are shown in Fig. 14. It can be seen that the feature

subsets selected by this algorithm have some advantages over the other three algorithms in the classification performance of FCN. When the SNR is 20dB, the highest classification accuracy of the FCN based on mRMR, FCBF and SRFS feature subset are 90.5%, 92.1%, and 93.9% respectively, which is lower than 95.8% of the proposed algorithm in this paper.

TABLE 1. Comparison of each selection algorithms.

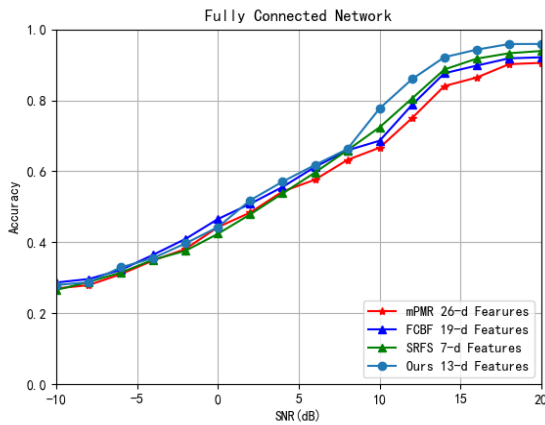| Algorithm	Feature number	Feature Subset	Time/s
mRMR	26	$f_2, f_3, f_5, f_6, f_7, f_8, f_9, f_{10}, f_{11},$ $f_{12}, f_{14}, f_{15}, f_{16}, f_{19}, f_{21}, f_{22}, f_{27},$ $f_{28}, f_{30}, f_{31}, f_{32}, f_{33}, f_{34}, f_{35}, f_{36}, f_{37}$	0.7s
FCBF	19	$f_3, f_6, f_7, f_8, f_9, f_{10}, f_{12},$ $f_{14}, f_{15}, f_{16}, f_{19}, f_{21}, f_{27},$ $f_{28}, f_{30}, f_{32}, f_{34}, f_{36}, f_{37}$	1.3s
SRFS	7	$f_3, f_9, f_{12}, f_{16}, f_{26}, f_{27}, f_{36}$	64.8s
This Paper	13	$f_3, f_7, f_8, f_9, f_{10}, f_{12}, f_{16}, f_{19}, f_{21}, f_{27}, f_{32}, f_{36}, f_{37}$	18.4s

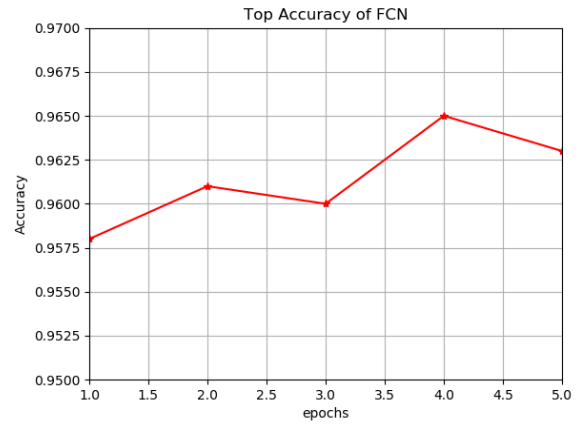**FIGURE 14.** Classification rate of different feature selection algorithms.

FIGURE 15. Top accuracy of FCN.

Finally, this section compares FCN with XGBOOST and KNN [33], 800 labeled samples with 13-dimensional optimal features are used to train them, then the convergence speed and the highest classification accuracy is shown in Table 2. XGBOOST compared in this paper belongs to the KERAS database.

It can be seen that the classification accuracy of FCN has some advantages over XGBOOST, KNN when the training time is basically same.

D. PSEUDO-LABEL LABELING

When the FCN training is finished, it is used to annotate the reliable unlabeled samples with pseudo-labels, then combined real-label samples and pseudo-label samples to retrain FCN until met the iteration condition, μ is set to 0.5 during network training. The top accuracy of FCN at 20dB is shown in fig 14. As can be seen from Fig. 15, the top classification accuracy of FCN does not increase significantly with the iteration and fluctuates around 96%, indicating the parameters of FCN are small and have been well fitted.

The samples number with iteration is shown in Table 3. It can be seen that the total number of pseudo-label samples does not reach 80000, which is due to the setting of reliable threshold. The proportion of number of each modulation
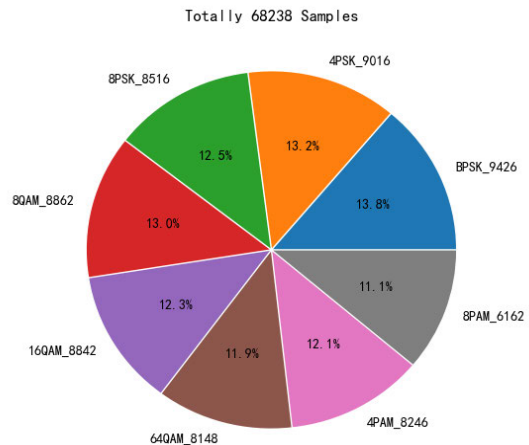

FIGURE 16. Number of different modulation types.

types is shown in Fig. 16. This lies in the fact that pseudo-label annotator can mark all kinds of signals to some extent and it is unlikely to happen that a certain type cannot be labeled.

E. CNN NETWORK TRAINING

Finally, all real-label samples and pseudolabel samples are sent into CNN for training, with a total of 68238 samples.
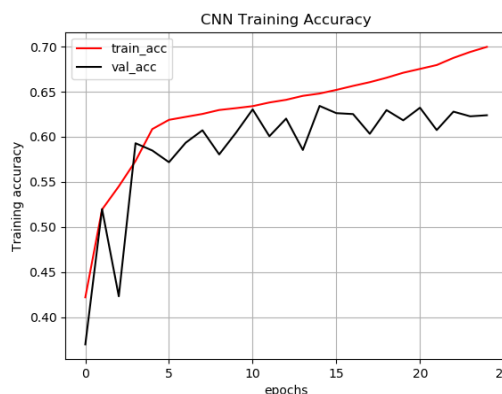
TABLE 2. Performance of different classifier.

CLASSIFIER	TOP ACCURACY	TRAINING TIME
FCN	95.8%	7.04s
XGBOOST	94.1%	6.26s
KNN	91.2%	8.31s

TABLE 3. Number of samples with iterations.

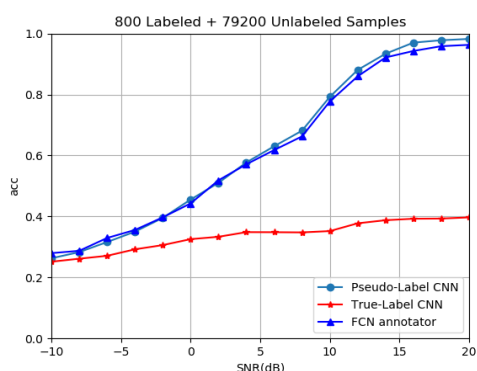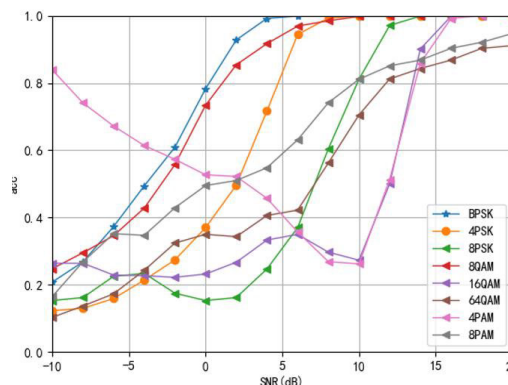| ITERATIONS	1	2	3	4	5
REAL-LABEL SAMPLES	800	800	800	800	800
PSEUDO-LABEL SAMPLES	67169	67417	67312	67475	67438
TOTAL SAMPLES	67969	68217	68112	68275	68238
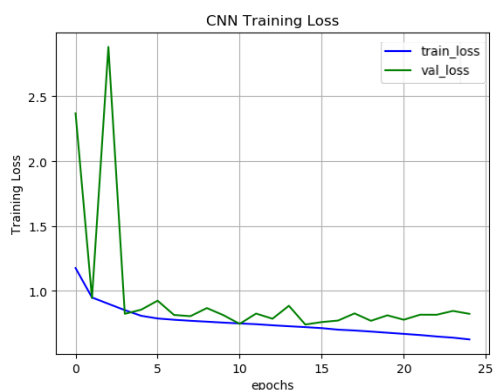
(a)CNN Training Curve

(b)CNN Accuracy Curve

FIGURE 17. CNN training accuracy and loss.

(a)Recognition Curve

(b)Recognition Rate of Each Signal

FIGURE 18. CNN classification accuracy.

Batchsize in the process of CNN training is set to 500, and 100 epochs for iterating. Crossentropy is used as the loss function, and Adam is used for optimization. The validation ratio is set to 0.1. In addition, a smaller learning rate is set as 0.001 to avoid model weights distorted too quickly. The performance of CNN training on training set and validation set is shown in Fig. 17, in which Fig. 17(a) shows the training loss of CNN and Fig. 17(b) shows the training accuracy of CNN. It can be seen that with the network continues to iterate, the training

accuracy and validation accuracy continues to increase and the training loss and validation loss continues to decrease. In order to prevent the network from overfitting, the early stopping condition is set such that the training stops when the validation loss does not decrease in 10 epochs. As a result, the training stopped at the 24th epoch when validation loss is stable.

It can be seen from Fig. 18(a) that after training CNN, the top classification accuracy reaches 98.3% when the SNR

is 20dB, which is still improved compared with 96.3% of the pseudo-label annotator. But if 800 real-label samples are used to train CNN merely, the classification accuracy of the CNN model is not ideal. Because the data size is too small to fit the CNN with a large number of parameters, the network has a serious problem of overfitting. The classification accuracy of each signal type at each SNR point is shown in Fig. 18(b).
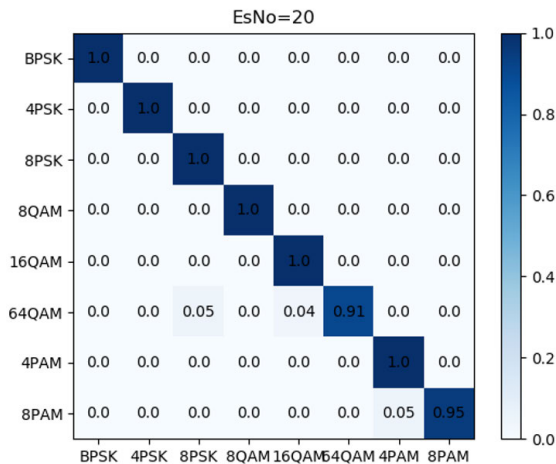

FIGURE 19. Confusion matrix at 20dB.

The confusion matrix at 20dB is shown in Fig. 19. Compared with Fig. 10(a), it can be seen that the classification accuracy of CNN for 64QAM and 8PAM has improved, which shows that the method based on deep learning is more effective when labeled samples are sufficient.
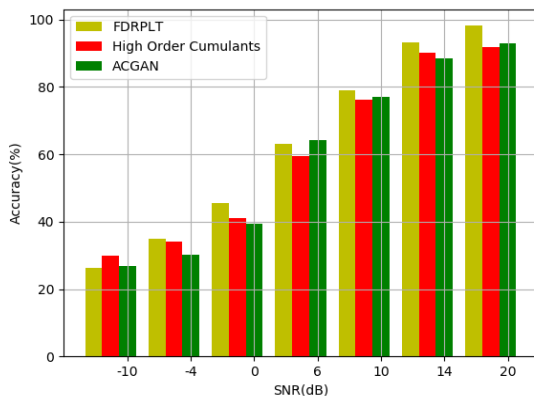

FIGURE 20. Comparison between pseudo-label CNN and other methods.

F. COMPARISON EXPERIMENT

In order to verify the overall performance of the algorithm in this paper, other methods about AMC are compared in this section. Fig. 20 shows the classification performance at different SNR in comparison with High Order Cumulants [34] and ACGAN [35].

Through the comparison results, we can see that FDRPLT has some advantages in classification accuracy when the SNR is high. In addition, the scheme proposed in this paper is simple and each module can be replaced flexible.

VII. CONCLUSION

This paper addresses the problem of few-shot modulation classification. We proposed a model named FDRPLT which combines the traditional handcrafted feature based method with the deep learning based method. First of all, according to the handcrafted feature based method, unsupervised learning based method and feature selection method, we generate a classifier with high classification performance under the constraint of labeled samples. Then we use the classifier to provide sufficient samples for CNN training. The simulation result shows that our scheme can achieve a good performance under the condition of small samples, when the SNR is 20 dB, the top classification accuracy of 8 modulation types can reach 98.3%.

Based on a lot of experiments, it is found that the performance of the model is closely related to the pseudo-label annotator. If the pseudo-label annotator can not raise the classification accuracy to a comparatively high level, the effective pseudo-label samples will obtain less, then the overall classification accuracy of deep neural network will descend. If the classification accuracy of the pseudo-label annotator can be further improved, the whole classification accuracy of the algorithm will be further improved, which is why our model have poor performance at low SNR. In this case if there is a better pseudo-label annotator or deep neural network, our FCN and CNN can be replaced flexibly. In the process of practical application, on one hand, we can monitor the distribution of pseudo-label samples, so as to know whether the deep learning method has enough samples for support. On the other hand, we can continuously improve the handcrafted feature set and unsupervised feature extraction algorithm, studying more discriminating features and selecting more complex models as pseudo-label annotator, so as to improve the performance of deep learning methods.

REFERENCES

[1] S. Kharbech, I. Dayoub, E. P. Simon, and M. Zwingelstein-Colin, "On classifiers for blind feature-based automatic modulation classification over multiple-input–multiple-output channels," *IET Commun.*, vol. 10, no. 7, pp. 790–795, May 2016.

[2] Z. Xing and Y. Gao, "Method to reduce the signal-to-noise ratio required for modulation recognition based on logarithmic properties," *IET Commun.*, vol. 12, no. 11, pp. 1360–1366, Jul. 2018.

[3] S. Ghasemi and A. Gangal, "An effective algorithm for automatic modulation recognition," in *Proc. 22nd Signal Process. Commun. Appl. Conf. (SIU)*, Trabzon, Turkey, Apr. 2014, pp. 903–906.

[4] A. E. Sherme, "A novel method for automatic modulation recognition," *Appl. Soft Comput.*, vol. 12, no. 1, pp. 453–461, Jan. 2012.

[5] H. Wang and L. Guo, "A new method of automatic modulation recognition based on dimension reduction," in *Proc. Forum Cooperat. Positioning Service (CPGPS)*, Harbin, China, May 2017, pp. 316–320.

[6] Z. Zhang, Y. Li, S. Jin, Z. Zhang, H. Wang, L. Qi, and R. Zhou, "Modulation signal recognition based on information entropy and ensemble learning," *Entropy*, vol. 20, no. 3, p. 198, Mar. 2018.

[7] H. Wang, L. Guo, Z. Dou, and Y. Lin, "A new method of cognitive signal recognition based on hybrid information entropy and D-S evidence theory," *Mobile Netw. Appl.*, vol. 23, no. 4, pp. 677–685, Aug. 2018.

[8] R. Zhou, F. Liu, and C. W. Gravelle, "Deep learning for modulation recognition: A survey with a demonstration," *IEEE Access*, vol. 8, pp. 67366–67376, 2020.

[9] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *Proc. Int. Conf. Eng. Appl. neural Netw.*, Aberdeen, Scotland, 2016, pp. 213–226.

[10] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 168–179, Feb. 2018.

[11] S. Jeong, U. Lee, and S. C. Kim, "Spectrogram-based automatic modulation recognition using convolutional neural network," in *Proc. 10th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Prague, Czech Republic, Jul. 2018, pp. 843–845.

[12] F. Meng, P. Chen, L. Wu, and X. Wang, "Automatic modulation classification: A deep learning enabled approach," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10760–10772, Nov. 2018.

[13] Z. Zhang, C. Wang, C. Gan, S. Sun, and M. Wang, "Automatic modulation classification using convolutional neural network with features fusion of SPWVD and BJD," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 5, no. 3, pp. 469–478, Sep. 2019, doi: 10.1109/TSIPN.2019.2900201.

[14] N. Gu, M. Fan, and D. Meng, "Robust semi-supervised classification for noisy labels based on self-paced learning," *IEEE Signal Process. Lett.*, vol. 23, no. 12, pp. 1806–1810, Dec. 2016.

[15] C.-Y. Chang, T.-Y. Chen, and P.-C. Chung, "Semi-supervised learning using generative adversarial networks," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Bengaluru, India, Nov. 2018, pp. 892–896.

[16] L. Cunhe and W. Chenggang, "A new semi-supervised support vector machine learning algorithm based on active learning," in *Proc. 2nd Int. Conf. Future Comput. Commun.*, Wuhan, China, 2010, pp. V3-638–V3-641.

[17] Y. Li, X. J. Guo, and X. Zhao, "Study on modulation recognition based on higher-order cumulants," *J. Southwest Univ. Sci. Technol.*, vol. 33, no. 3, pp. 64–68, 2018.

[18] A. Wang and R. Li, "Research on digital signal recognition based on higher order cumulants," in *Proc. Int. Conf. Intell. Transp., Big Data Smart City (ICITBS)*, Changsha, China, Jan. 2019, pp. 586–588.

[19] Z. Zhao, A. Yang, and P. Guo, "A modulation format identification method based on information entropy analysis of received optical communication signal," *IEEE Access*, vol. 7, pp. 41492–41497, 2019.

[20] A. Ali and F. Yangyu, "Automatic modulation classification using deep learning based on sparse autoencoders with nonnegativity constraints," *IEEE Signal Process. Lett.*, vol. 24, no. 11, pp. 1626–1630, Nov. 2017.

[21] T. Ya, Y. Lin, and H. Wang, "Modulation recognition of digital signal based on deep auto-ancoder network," in *Proc. IEEE Int. Conf. Softw. Qual., Rel. Secur. Companion*, Prague, Czech Republic, Jul. 2017, pp. 256–260.

[22] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Unsupervised representation learning of structured radio communication signals," in *Proc. 1st Int. Workshop Sens., Process. Learn. Intell. Mach. (SPLINE)*, Aalborg, Denmark, Jul. 2016, pp. 1–5.

[23] E. Kodirov, T. Xiang, and S. Gong, "Semantic autoencoder for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 4447–4456.

[24] M. Waqar Aslam, Z. Zhu, and A. K. Nandi, "Automatic modulation classification using combination of genetic programming and KNN," *IEEE Trans. Wireless Commun.*, vol. 11, no. 8, pp. 2742–2750, Aug. 2012.

[25] R. S. B. Krishna and M. Aramudhan, "Feature selection based on information theory for pattern classification," in *Proc. Int. Conf. Control, Instrum., Commun. Comput. Technol. (ICCICCT)*, Kanyakumari, India, Jul. 2014, pp. 1233–1236.

[26] Y. Han, Y. Yang, Y. Yan, Z. Ma, N. Sebe, and X. Zhou, "Semisupervised feature selection via spline regression for video semantic recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 2, pp. 252–264, Feb. 2015.

[27] K. Benabdeslem and M. Hindawi, "Efficient semi-supervised feature selection: Constraint, relevance, and redundancy," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1131–1143, May 2014.

[28] J. C. Ang, A. Mirzal, H. Haron, and H. N. A. Hamed, "Supervised, unsupervised, and semi-supervised feature selection: A review on gene selection," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 13, no. 5, pp. 971–989, Sep. 2016.

[29] L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," *J. Mach. Learn. Res.*, vol. 5, no. 12, pp. 1205–1224, 2004.

[30] Y. Wang, J. Wang, and H. Liao, "An efficient semi-supervised representatives feature selection algorithm based on information theory," *Pattern Recognit.*, vol. 61, no. 5, pp. 511–512, 2017.

[31] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.

[32] M. Bouchou, H. Wang, and M. El Hadi Lakhdari, "Automatic digital modulation recognition based on stacked sparse autoencoder," in *Proc. IEEE 17th Int. Conf. Commun. Technol. (ICCT)*, Chengdu, China, Oct. 2017, pp. 28–32.

[33] S. Zhang, X. Li, M. Zong, X. Zhu, and D. Cheng, "Learning K for KNN classification," *ACM Trans. Intell. Syst. Technol.*, vol. 8, no. 3, pp. 1–19, 2017.

[34] W. Xie, S. Hu, C. Yu, P. Zhu, X. Peng, and J. Ouyang, "Deep learning in digital modulation recognition using high order cumulants," *IEEE Access*, vol. 7, pp. 63760–63766, 2019.

[35] 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.

YUNHAO SHI received the B.S. degree in information engineering from Xidian University, Xi'an, China, in 2018. He is currently pursuing the M.S. degree with the Information and Navigation College, Air Force Engineering University, Xi'an. His research interests include deep learning, signal processing, and few-shot learning.

HUA XU received the B.S. and M.S. degrees in communication engineering from Air Force Engineering University, Xi'an, China, in 2001, and the Ph.D. degree in communication signal processing from Information Engineering University, Zhengzhou, China, in 2005. He is currently a Professor with Air Force Engineering University. His research interests include communication signal processing, blind signal processing, and communication countermeasures.

LEI JIANG received the B.S., M.S., and Ph.D. degrees in communication engineering and communication signal processing from Air Force Engineering University, Xi'an, China, in 2005. He is currently an Associate Professor with Air Force Engineering University. His research interests include communication systems, electronic countermeasures, and pattern recognition.

YINGHUI LIU received the B.S. degree in electronic and information engineering from Xidian University, Xi'an, China, in 2018. He is currently pursuing the M.S. degree with the Information and Navigation College, Air Force Engineering University, Xi'an. His research interests include deep learning and signal processing.

● ● ●