IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 8, 2019, accepted July 16, 2019, date of publication July 22, 2019, date of current version August 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2930250

Radar Waveform Recognition Based on Multiple Autocorrelation Images

ZHI HUANG, ZHIYUAN MA, AND GAOMING HUANG
Department of Electronic Technology, Naval University of Engineering, Wuhan 430033, China

Corresponding author: Zhiyuan Ma (mazhiyuan@mails.ccnu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61501484

ABSTRACT Radar signal waveform recognition, as a key component of radar target recognition, has always been a research topic of great concern in the field of electronic countermeasures. In this paper, aiming at the contradiction between improving recognition rate and reducing the sample size, we propose a multiple autocorrelation joint decision models, which achieves higher waveform recognition rate while requires lower data volume for the original sampled data. The key point of the model is to perform multiple autocorrelations on the signal and use time-frequency transform for each times autocorrelation result to obtain multiple time-frequency feature images that can characterize the same signal. Then, the model adapts to the input of multiple feature images and gets the pre-classification results of each feature image. Finally, using the pre-classification results, this paper designs an inference machine module based on a fully connected structure to achieve a better signal classification result. This paper simulates six types of the signals and generates training sets and test sets, using two data sets to achieve hyper parameter optimization, training, and testing of the model. The simulation results compared with the literature show that the proposed model not only has a high recognition rate at a high signal-to-noise ratio (SNR) but also better adapts to waveform recognition at low SNR environment. At −9dB SNR, the recognition rate of six types of signals is more than 74%.

INDEX TERMS Autocorrelation, inference machine, low SNR, radar signal waveform recognition.

I. INTRODUCTION

Effective radar signal waveform recognition (RSWR) is an important support of the electronic warfare (EW) systems, such as electronic intelligence (ELINT), electronic support measure (ESM), electronic protection (EP), and electronic attack (EA) systems. In practice, based on the RSWR, the warfighters mainly complete the interception, localization, analysis and recognition of enemy radar signals, and provide commanders with battlefield situation information and necessary tactical actions [1]. Therefore, the radar receiver should have the ability to accurately recognize the radar signal.

In current research, waveform recognition usually consists of two parts: feature extraction and classifier design. For feature extraction stage, because radar signals are usually non-stationary signals, time-frequency analysis is widely used in feature extraction, such as Choi-Williams distribution (CWD) [2]–[5], short-time Fourier transform (STFT) [6]–[8], smoothed pseudo-Wigner distribution (SPWD) [9], and Wigner Ville distribution (WVD) [10]. At the same time, some theories use other feature extraction methods, such as feature extraction using IQ channel signal waveform in time domain [11], principal component analysis (PCA) [12], radon ambiguity transform and radon-WVD [13], and multi-domain feature extraction [14], [15]. In [14] and [15], they extract the time-domain and time-frequency domain features of signals. In time-domain, they extract signal features based on power spectral density, statistics and instantaneous properties. In time-frequency domain, they extract image features based on binarization images. In addition, some novel signal features have been proposed, such as ZAM-GTFR [16] and spatial features [17]. At the stage of classifier design, classification methods include support vector machine (SVM) [9], [15], [17], convolutional neural networks (CNN) [4], [5], [10], Elman neural networks (ENN) [3], multi-layer perceptron (MLP) [2], and conditional judgment of various features [8], [13]. There are also hybrid classifiers that combine multiple classifiers. For example, in [12], the classifier is composed of k-Nearest Neighbor (KNN), random forest and neural network (NN). In [11], [14], and [18], they all combine two deep learning models, in [14],

The associate editor coordinating the review of this manuscript and approving it for publication was Sudhakar Radhakrishnan.

the classifier is composed of CNN and ENN, in [18], the classifier is composed of CNN and SAE, and in [11], the classifier is composed of a two-channel CNN combining with Bi-directional Long Short-Term Memory (TCNN-BL).

In recent years, because of the important breakthroughs made by CNN in the field of image classification [19], researchers more frequently use deep learning classifiers to achieve waveform classification [3]–[6], [10], [11]. We have noticed that time-frequency analysis can extract features from target signals from the two aspects of signal processing and image processing, and the two-dimensional data generated by time-frequency analysis can better meet the requirements of CNN for input data, so it is widely used in feature extraction stage [2]–[10]. However, the time-frequency image (TFI) is easily affected by noise at low SNR. In order to extract the effective features of the signal in the time-frequency domain, the signal needs to be denoised to achieve better feature extraction and waveform classification. In [5], the sample averaging technique (SAT) is used to suppress noise, which improves the SNR of 6dB in the TFI, and the recognition rate of twelve types of signals reaches 85% at −16dB. In [13], it combines Hilbert transform, instantaneous autocorrelation, non-linear wavelet transform and WVD to suppress noise. The recognition rate of eight types of signals reaches 95% at −10dB. But in order to extract the time-frequency characteristics of the signal, [5] and [13] inevitably require more sampling data due to the need for noise reduction algorithms, which need 10,000 sample points for each recognition. In order to suppress the noise in time-frequency analysis and reduce the sample size, some literatures use the data in a pulse of the signal to filter the noise from the perspective of image processing. In [3], [4], [14], and [15], they form a TFI and perform filtering processing by image threshold denoising and image morphology. The reference [9] uses image enhancement, image threshold binarization, and image morphology to process noise contained in TFIs. In [10], the TFI is updated by smoothing filter. And in [6], a stacked convolutional denoising auto-encoders (SCDAEs) is proposed, which reconstructs the original data to further remove the impact of noise in TFIs. It is noted that the denoising effect of image processing in the TFI is not very obvious in the above-mentioned literature, especially when the SNR is less than −6dB, because noise almost submerges the signal in the TFI, the image denoising becomes very difficult. As a result, in [3], [4], [14], and [15], their better recognition performance requires a SNR greater than −5dB. In [6],the recognition rate of six types of signals is 60% at −8dB. We realize that due to the sensitivity of TFIs to noise, if we want to improve the recognition rate of signals, more sampling data must be acquired to facilitate signal de-noising, such as [5], [13]. The small samples used to reduce the requirement of raw data mean the sacrifice of recognition rate, such as [3], [6], [14], and [15]. Therefore, it's difficult to trade off the recognition rate of signals and the size of the original sampling data.

In fact, in order to obtain high recognition rate with small samples, the key point of combining signal image and deep learning to achieve waveform classification is to obtain a feature image that can effectively characterize the signal, which should be less susceptible to noise. In [21], note that autocorrelation is a method for effectively detecting signals in a weak signal environment, while time-frequency analysis can achieve data transformation from one-dimensional to two-dimensional [23], which allows us to obtain a new feature image. Therefore, in order to solve the problem that TFIs at low SNR environment are susceptible to noise, this paper designs a novel feature image combined with autocorrelation and time-frequency analysis, which has better anti-noise performance and can robustly represent the target signal at low SNR. Using this feature image and the network structure designed in this paper, the way that we mitigate the contradiction between improving the recognition rate and reducing the sample size is further improving the waveform classification accuracy under the sampling data in one pulse of the signal.

In this paper, we proposes a multiple autocorrelation joint decision (MAJD) model, which consists of three parts: the data pre-process, the network, and the inference machine module. Firstly, the signal is processed by the data pre-processing module proposed in this paper. Data pre-process will make multiple autocorrelation of signals. Similar to the literature [3], [6], [14], and [15], the time-frequency transform of the autocorrelation signal will be further carried out to adapt to the dimensional requirements of the CNN on the input data. Secondly, the CNN is used to automatically extract the deep essential features of two-dimensional data. In order to better extract the deep features of the TFI, four groups of CNNs are constructed to extract signal features simultaneously, and each group of CNNs consists of 11 layers including convolutional layer, pooling layer and fully connected layer. Thirdly, the pre-classification result obtained by the CNN is further input into the inference machine, which judges the probability that the pre-classification result may be the final classification result. Finally, we obtains the possible signal class of the input signal through the inference machine.

This paper is organized as follows. Section II introduces the basic framework of RSWR and the basic structure of the CNN. Section III presents the data pre-processing method and the MAJD model designed according to the data pre-process. Section IV gives the simulation experiment and results, which describes the data set generation method of the simulation experiment and the simulation results of the model proposed in this paper. Section V gives the simulation conclusion.

II. BASIC THEORY FOR THE RSWR

In this section, we present the basic framework of RSWR. In order to describe the network model proposed in this paper, the structure of the CNN is modelled in this section.

A. BASIC FRAMEWORK OF RECOGNITION

The basic task of RSWR is to classify various types of radiation source signals in the space. Specifically, as shown in Fig. 1, the signal in space is received by the antenna

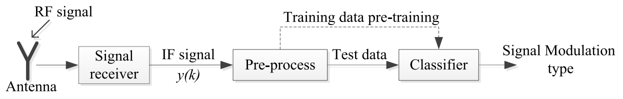


FIGURE 1. **The framework of radar signal waveform recognition.**

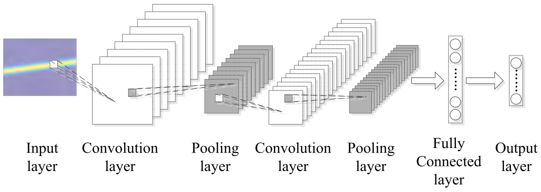


FIGURE 2. **The Structure of CNN.**

into the signal receiver, and the received RF signal is mixed and sampled at the receiver to output an IF sampling signal $y(k)$. The IF sampling signal will be pre-processed to reduce the influence of spatial noise on the recognition task. The pre-process may be performed directly on the IF sampling signal $y(k)$, or the signal may be transferred to the transform domain for pre-processing. In [4], [6], and [11], they carries out signal denoising processing in the time-frequency transform domain. In [5], it combines two pre-processing stages, firstly carries out SAT on the signal, and then pre-processes the TFI of the signal in the time-frequency domain. We also combine two pre-processing stages. Firstly, the signal is processed by multiple autocorrelation, and then the signal is converted to the time-frequency domain. Secondly, the feature image of the signal is sampled down, which can reduce the sample size and the memory consumption of GPU and CPU. Finally, the pre-processing signals are input into the classifier which designed and pre-trained in this paper to complete the classification of signal modulation type. Similar to the previous literature [3]–[5], [13], and [20], we focus on the classification of modulation type and assume that the sampling data within a pulse of the signal is complete.

B. CONVOLUTIONAL NEURAL NETWORK STRUCTURE

This paper designs classifier based on the CNN. The classical structure usually consists of five parts: input layer, convolution layer, pooling layer (i.e., sampling layer), fully connected layer, and output layer. Feature extraction and dimension reduction are accomplished by convolution and pooling. We use time-frequency analysis method to convert the signal into the feature image (i.e., time-frequency image), and finally divide it into target types through the fully connected layer, as shown in Fig. 2. We choose S^{n}_{ij} to denote the j-th feature map of the i-th pooling layer, and the feature map size is $n \times n$. The C^{n}_{ik} denotes the k-th feature map of the i-th convolution layer, and the feature map size is $n \times n$. The W^{l}_{ikj} denotes the j-th convolution kernel of the k-th group convolution kernel in the i-th convolution layer, and the size of the convolution kernel is $l \times l$. When $i = 0$ in S^{n}_{ij}, the S^{n}_{0j} denotes the input image. We need to emphasize that since the time-frequency feature image is a grayscale image, the input layer has only one channel, which means that the input image has only one feature map. When the CNN carries out image recognition, the convolution and pooling operations are executed alternately.

In the operation of convolution, the k-th feature map of the $(i+1)$-th convolution layer can be calculated as follows

$$C^{n}_{(i+1)k} = \sum_{j} (S^{n}_{ij} \otimes W^{l}_{(i+1)kj}) + b_k \tag{1}$$

where ”$\otimes$” is a convolution operation. The b_k is a convolution kernel bias of the k-th group.

After getting the k-th feature map of the $(i+1)$-th convolution layer, the CNN continues to do the pooling operation

$$S^{n/2}_{(i+2)k} = Sample(C^{n}_{(i+1)k}) \tag{2}$$

where $Sample(\cdot)$ is a pooling function. The size of the pooling area is set as 2×2 in this paper, and the pooling mode is the maximum pooling. After pooling, the number of the feature maps remains unchanged, but the size is changed to $(n/2) \times (n/2)$.

After convolution and pooling, the final feature map is expanded into a set of one-dimensional vectors, which are then divided into m target classes through several fully connected layers

$$[y_1, y_2, \ldots, y_{m-1}, y_m] = Fullconnect(S^{n}_{ij}) \tag{3}$$

where $Fullconnect(\cdot)$ is a fully connected function.

In [22], the author has already given the specific calculation methods of convolution, pooling and fully connected, so we do not introduce the detailed calculation.

III. SIGNAL PRE-PROCESSING

In this section, aiming at the problem that TFIs are susceptible to noise, we design a novel feature image using data in one pulse of signals combined with autocorrelation and time-frequency analysis. The feature image is less susceptible to noise and is capable of uniquely characterizing the signal. Finally, In order to overcome the influence of the similarity among feature images, we use multiple autocorrelation to obtain multiple feature images that can characterize the signal.

A. THE RADAR SIGNAL

Generally, the output signal of the receiver can be expressed as

$$y(k) = x(k) + n(k) = A(k)e^{j\theta(k)} + n(k) \tag{4}$$

where $x(k)$ is the ideal discrete signal after IF sampling, $n(k)$ is additive white Gaussian noise, k is an index value that increases sequentially with the sampling interval, $A(k)$

TABLE 1. Parameters of signals.

Modulation type	$f(k)$ [HZ]	$\phi(k)$ [rad]
CP	f_c	constant
LFM	$f_c + \frac{B}{\tau_{pw}}(kT_s)$	constant
NCPM	f_c	$f_0 - f_k \sin 2\pi f_k (kT_s)$
BPSK	f_c	0 or π
BFSK	f_1、f_2	constant
QFSK	f_1、f_2、f_3、f_4	constant

is the instantaneous envelope of the ideal sampling signal, $\theta(k)$ is the instantaneous phase of the ideal sampling signal. To explain further, the relationship between instantaneous phase $\theta(k)$, instantaneous frequency $f(k)$ and instantaneous phase offset $\phi(k)$ can be described as

$$\theta(k) = 2\pi f(k)(kT_s) + \phi(k) \tag{5}$$

where, T_s is the sampling interval of the signal. We simulate six types of radiation source signals, which are conventional pulse (CP), linear frequency modulation (LFM), nonlinear cosine phase modulation (NCPM), binary phase shift keying (BPSK), binary frequency shift keying (BFSK), and quaternary frequency shift keying (QFSK). The composition of instantaneous frequency and instantaneous phase offset of the six types of radiation source signals used in this simulation are showed in Table 1. And the instantaneous envelope is set to $A(k) = 1$.

As shown in Table 1, carrier frequencies f_1, f_2, f_3, f_4 and f_c are fixed values. In LFM, B is the signal bandwidth, τ_{pw} is the signal pulse width, and kT_s is the discrete sampling time. In NCPM, f_k is the modulation frequency of its instantaneous phase.

B. FEATURE EXTRACTION IN TIME-FREQUENCY DOMAIN

In this paper, time-frequency analysis is used to extract two-dimensional features of signals. The general time-frequency analysis method with bilinear form is expressed as follows

$$C_x(t, \Omega|g) = \frac{1}{2\pi} \int\int\int x(u + \tau/2)x^*(u - \tau/2) \times g(\theta, \tau)e^{-j(\theta t + \Omega\tau - u\theta)} du d\tau d\theta \tag{6}$$

where $g(\theta, \tau)$ is the kernel function of time-frequency analysis. By changing the parameter in $g(\theta, \tau)$, we can acquire different time-frequency distribution. Among many time-frequency distribution methods, the CWD can effectively suppress the time-frequency cross-term by adjusting the parameters. Therefore, this paper chooses the CWD as the time-frequency analysis method. The kernel function composition of the CWD is as follows

$$g(\theta, \tau) = e^{-j\theta^2\tau^2/\sigma} \tag{7}$$

The final expression of the CWD is

$$C_x(t, \Omega) = \int\int \sqrt{\frac{\pi\sigma}{\tau^2}} x(u + \tau/2)x^*(u - \tau/2) \times e^{-\pi^2\sigma(u-t)^2/4\tau^2 - j\Omega\tau} du d\tau \tag{8}$$

where, $C_x(t, \Omega)$ is the result of time-frequency analysis [23], t is the time axis, and Ω is the frequency axis. Scale factor σ is used to control the cross term generated by the CWD. The frequency resolution of the CWD decreases when σ is used to suppress cross terms. In this paper, we use $\sigma = 1$ to balance cross terms and frequency resolution, and use $CWD(\cdot)$ to denote Choi-Williams transform for input signals. The result of time-frequency transform can be expressed as

$$[\boldsymbol{t}, \boldsymbol{f}, \boldsymbol{trf}] = CWD(y(k)) \tag{9}$$

where the one-dimensional vector $\boldsymbol{t} \in \mathbb{R}^n$, two-dimensional vector $\boldsymbol{f} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{trf} \in \mathbb{R}^{n\times n}$ are obtained by the CWD. And each element in $\boldsymbol{t}$ denotes the time, each element in $\boldsymbol{f}$ denotes the normalized frequency, and each element in $\boldsymbol{trf}$ is composed of a number to denote strength. In order to convert $\boldsymbol{trf}$ into a two-dimensional image, considering that the pixel value range of the image is 0 to 255, $\boldsymbol{trf}$ needs to be transformed into the pixel range. For the convenience of expression, $\boldsymbol{trf}_{ij}$ is used to denote the value of the i-th row and the j-th column of the two-dimensional matrix $\boldsymbol{trf}$

$$pic = 255\frac{\boldsymbol{trf}_{ij} - \min(\boldsymbol{trf})}{\max(\boldsymbol{trf}) - \min(\boldsymbol{trf})} \tag{10}$$

where $\max(\cdot)$ and $\min(\cdot)$ denote the maximum value function and the minimum value function, and pic denotes the two-dimensional image formed by mapping the two-dimensional matrix to the pixel interval. For the convenience of the following statement, we use function $map(\cdot)$ to denote this mapping method.

We simulate six types of radiation source signals through the CWD, and their TFIs are shown in Fig. 3.

Using the TFIs of the signal, the reference [3], [14], and [15] had achieved signal waveform classification. However, the TFI is susceptible to noise, especially at low SNR, which are almost indistinguishable, as shown in Fig. 4(a). In order to reduce the influence of noise on TFI, [3], [14], and [15] preprocess TFI by image threshold denoising, but this method is still sensitive to noise, as shown in Fig. 4(b). In fact, the key to signal classification using images is to obtain a feature image that can effectively characterize the signal, which should be less susceptible to noise. However, most of the current literatures implement signal classification using TFIs, which leads to poor results of signals at low SNR. Considering this requirement, we use autocorrelation and multiple autocorrelation design a novel feature image that can represent the signal. The feature image has better anti-noise ability, and the pixel characteristics of the signal are more obvious at low SNR. The feature image formed by autocorrelation is shown in Fig. 4(c), and the feature image formed by double autocorrelation is shown in Fig. 4(d). Compared with the original TFI in Fig. 4(a), the autocorrelation

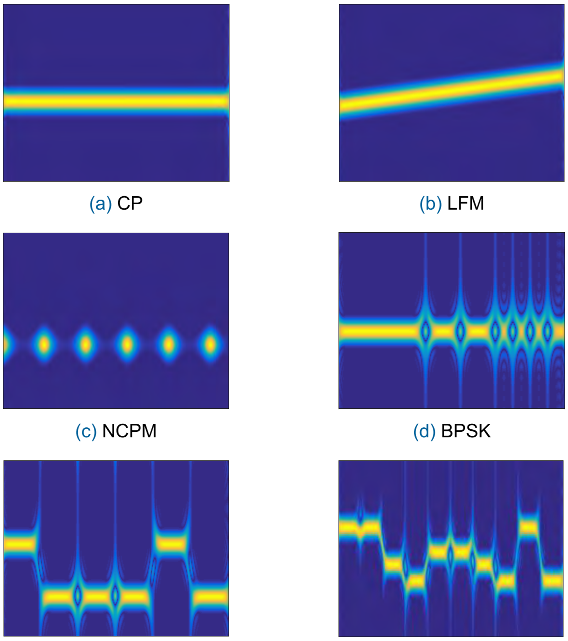


FIGURE 3. Time-frequency images of six types of radiation source signals for noise-free environment.

image in Fig. 4(c) has a clearer distribution of pixels, while the multiple autocorrelation image in Fig. 4(d) has a higher intensity of pixels. This kind of image undoubtedly provides more accurate information of signal characteristics, which is more conducive to signal classification. In section III-C and III-D, We will further elaborate on the construction of the feature image.

C. SIGNAL AUTOCORRELATION

Signal autocorrelation can effectively filter the noise in the signal. Specifically, given signal $x(t)$ and zero-mean Gaussian white noise $n(t)$, the observable signal is $y(t) = x(t) + n(t)$, then the autocorrelation function of $y(t)$ can be expressed as

$$R_y(\tau) = E[y(t)y(t-\tau)] \tag{11}$$

So $R_y(\tau)$ can be also denoted as

$$\begin{aligned} R_y(\tau) &= E[y(t)y(t-\tau)] \\ &= E\{[x(t)+n(t)][x(t-\tau)+n(t-\tau)]\} \\ &= E[x(t)x(t-\tau)] + E[n(t)n(t-\tau)] \\ &\quad + E[x(t)n(t-\tau)] + E[n(t)x(t-\tau)] \\ &= R_x(\tau) + R_n(\tau) + R_{xn}(\tau) + R_{nx}(\tau) \end{aligned} \tag{12}$$

Considering that noise $n(t)$ is unrelated to signal $x(t)$, then $R_{xn}(\tau) = R_{nx}(\tau) = 0$, so

$$R_y(\tau) = R_x(\tau) + R_n(\tau) \tag{13}$$

For the zero-mean Gaussian white noise $n(t)$ with wide bandwidth, the autocorrelation function $R_n(\tau)$ mainly reacts near $\tau = 0$. When τ is large, $R_y(\tau)$ mainly reacts to $R_x(\tau)$. At this time, $R_y(\tau) \approx R_x(\tau)$. Since $R_x(\tau)$ is determined only by signal $x(t)$, $R_y(\tau)$ is also determined by $x(t)$. For convenience of description, the obtained $R_y(\tau)$ is denoted as $R_y(t)$, combined with time-frequency transform, $R_y(t)$ is substituted into (8) to obtain time-frequency matrix, which is expressed as

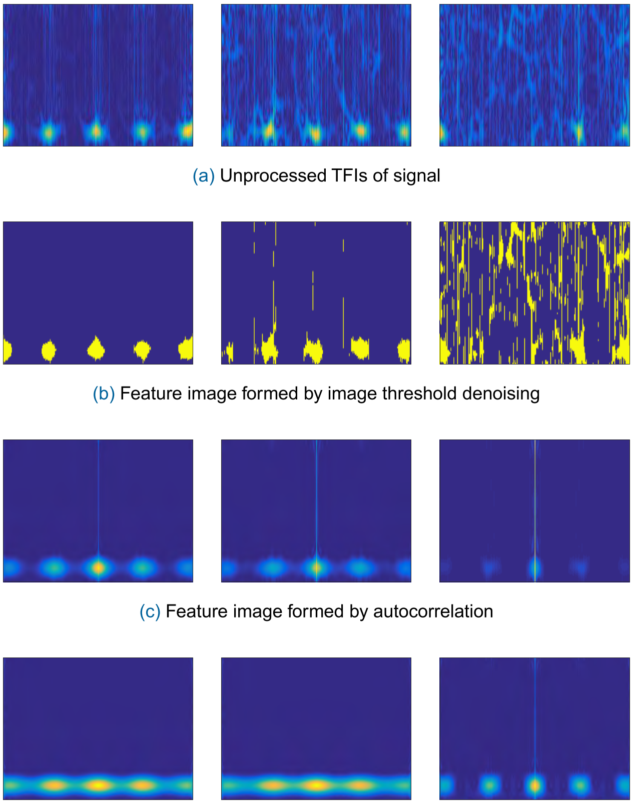


FIGURE 4. Image formed by various signal preprocessing methods. Feature images formed from left to right when SNR is 0dB, −5dB and −10dB, respectively.

$$C_R(t,\Omega) = \int\int \sqrt{\frac{\pi\sigma}{\tau^2}} R_y(u+\tau/2)R_y^*(u-\tau/2) \; e^{-\pi^2\sigma(u-t)^2/4\tau^2 - j\Omega\tau} du d\tau \tag{14}$$

According to (13), the result based on CWD reflects the change of autocorrelation function $R_y(t)$. Since $R_y(t)$ is only determined by signal $x(t)$, $C_R(t, \Omega)$ is the reaction of $x(t)$ in two-dimensional time-frequency plane after passing through autocorrelation domain, which means that it can uniquely represent signal $x(t)$.

The discrete value $y(k)$ is obtained after IF sampling. We calculate the autocorrelation function of $y(k)$ with (15) and (16)

$$\hat{H}_y(k) = \begin{cases} \sum_{n=0}^{N-1} y(n)y(n-k) & k \geq 0 \\ \hat{H}_y(-k) & k < 0 \end{cases} \tag{15}$$

According to the value of $\hat{H}_y(k)$, the autocorrelation sequence $\hat{R}_y(k)$ is further obtained.

$$\hat{R}_y(k) = \hat{H}_y(k-N) \quad k = 1, 2, \ldots, 2N-1 \tag{16}$$

where N denotes the number of sampling points, $\hat{R}_y(k)$ denotes the autocorrelation value obtained according to the actual value of sampling points, when $n < 0$, $y(n) = 0$. Combining time-frequency analysis and utilizing the special autocorrelation value distribution of $\hat{R}_y(k)$, we can obtain the feature image which can represent the signal.

Combined with signal autocorrelation and time-frequency analysis, we simulate six types of radiation source signals in a SNR environment of 9dB, and obtained their two-dimensional images as shown in Fig. 5. It can be found that although the combination of autocorrelation and time-frequency analysis can form autocorrelation images, the pixel characteristics of each image are not very obvious, and some signals are even a little similar in image features, so the autocorrelation images obtained by this way need further processing in section III-D.

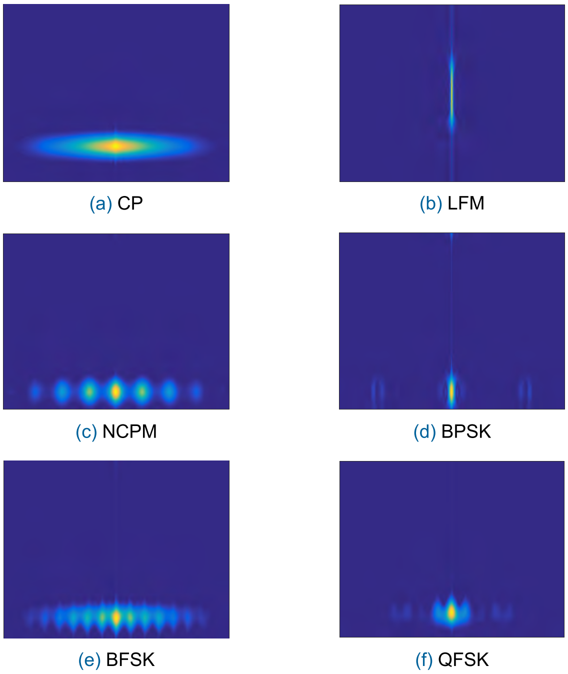

(a) CP
(b) LFM
(c) NCPM
(d) BPSK
(e) BFSK
(f) QFSK

FIGURE 5. Time-frequency image formed in a SNR environment of 9dB by autocorrelation and time-frequency analysis.

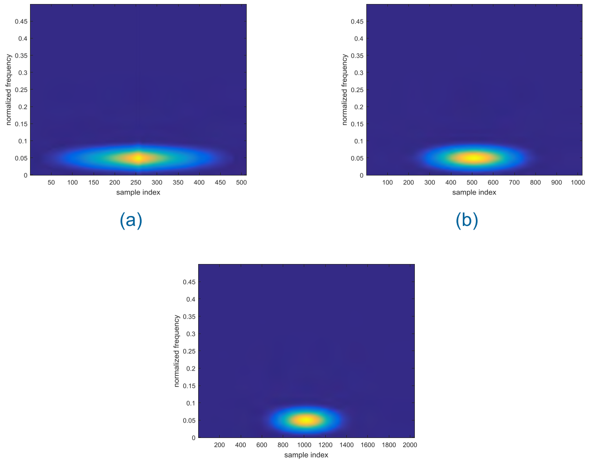
D. SIGNAL MULTIPLE AUTOCORRELATION

Signal multiple autocorrelation can effectively enhance the intensity of useful pixels in an image, which allows us to obtain a feature image that accurately represents the signal. The signal multiple autocorrelation will calculate the autocorrelation results once more, i.e. iteratively calculate several times autocorrelation. In this paper, $acorr(\cdot)$ is used to denote the autocorrelation function composed of (15) and (16). $y_n(k)$ is the result of n times autocorrelation, where n is the natural number. Then the multiple autocorrelation can be expressed as

$$y_{n+1}(k) = acorr(y_n(k)) \tag{17}$$

When $n = 0$, $y_0(k)$ denotes the original signal $y(k)$. After n times autocorrelation, we can obtain the autocorrelation sequences $y_1(k), y_2(k), \ldots, y_n(k)$. Next, we transform $y_1(k), y_2(k), \ldots, y_n(k)$ into time-frequency domain, and obtain n feature images which can represent signals.

However, assuming that the length of the original signal is N, then according to (15) and (16), the length of $y_n(k)$ can be expressed as $2^nN - 2^n + 1$, so the length of $y_n(k)$ obtained through multiple autocorrelation will increase rapidly, which will lead to the enlargement of feature images and increase the computational complexity. In order to reduce the size of feature images and the memory consumption of GPU and CPU, we further clip $y_n(k)$. Considering (15) and (16), since the values at both ends of the autocorrelation are calculated by less discrete values, there is almost no valid pixel information at the edge of feature images. Fig. 6 shows the change of the feature image of the CP after triple autocorrelation. It can be found that at the edge of feature images, their pixel strength gradually weakens until they disappears.



(a)
(b)
(c)

FIGURE 6. The effect of multiple autocorrelation on the CP. (a), (b), and (c) respectively represent the feature image of the CP obtained by one, two, and three times autocorrelation combined with time-frequency analysis.

It can also be seen from Fig. 6 that as the length of $y_n(k)$ increases, the feature image becomes larger, while the pixel area which can be used to represent the signal does not change. Therefore, when $n > 0$, we take the interval $(floor(\frac{2^nN-2^n+1}{2})-\frac{N}{2}, floor(\frac{2^nN-2^n+1}{2})+\frac{N}{2})$ of $y_n(k)$ to form the autocorrelation value $y'_n(k)$ with the same length of N, and $floor(\cdot)$ denotes the rounding down function. In this way, the length of the calculated results from multiple autocorrelation is always N. We use $clip(\cdot)$ to denote the relationship between $y_n(k)$ and $y'_n(k)$, and they can be expressed as

$$y'_n(k) = clip(y_n(k)) \tag{18}$$

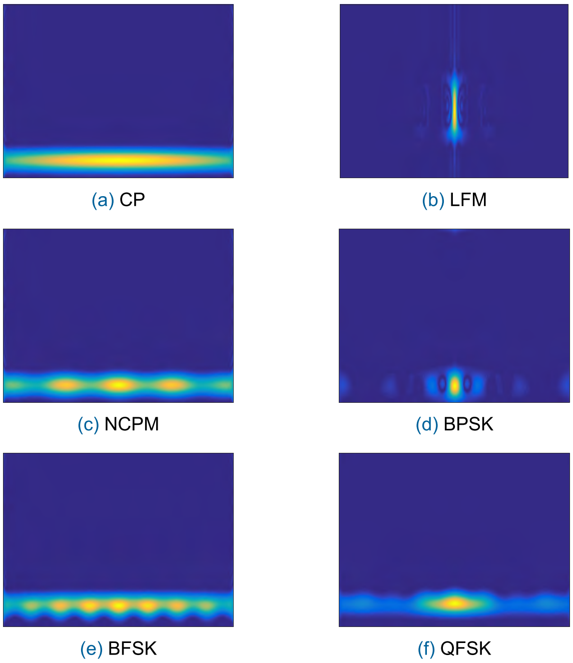

(a) CP (b) LFM (c) NCPM (d) BPSK (e) BFSK (f) QFSK

FIGURE 7. Time-frequency images formed by double autocorrelation, and the SNR is 9dB.

After $clip(\cdot)$, we transform $y_2'(k)$ into time-frequency domain and get the feature images of six types of signals, as shown in Fig. 7, Comparing with Fig. 5, we can find that the multiple autocorrelation processing enables the feature images to fully reflect the overall change of the signal after passing through the autocorrelation domain, while removing the invalid part of the image. Based on this method, we finally get the autocorrelation feature image of the signal.

Although autocorrelation images avoid the problem that TFIs are susceptible to noise, there are still some problems in autocorrelation images. For example, in Fig. 7, the feature images of CP, NCPM, BFSK, and QFSK are similar. We cannot ignore this similarity, because it will affect the classification of signals. In order to overcome the impact of this similarity, we hope to obtain more features that can reflect the signal, which will further improve our classification accuracy.

In fact, according to Fig. 7, the feature image obtained by multiple autocorrelation can improve the intensity of the pixels of the image effectively characterizing the signal, which means that the image formed by multiple autocorrelation can also be used for signal classification. If we can obtain several feature images by the signal multiple autocorrelation, it will provide more initial information for the classification of networks, thereby improving the accuracy of the network for classifying signal waveforms. According to the analysis in this section, n times autocorrelation can obtain n feature images. Therefore, we try to find a reasonable number of autocorrelation times to obtain more feature images that can represent signals. It will be introduced in detail in the section III-E.

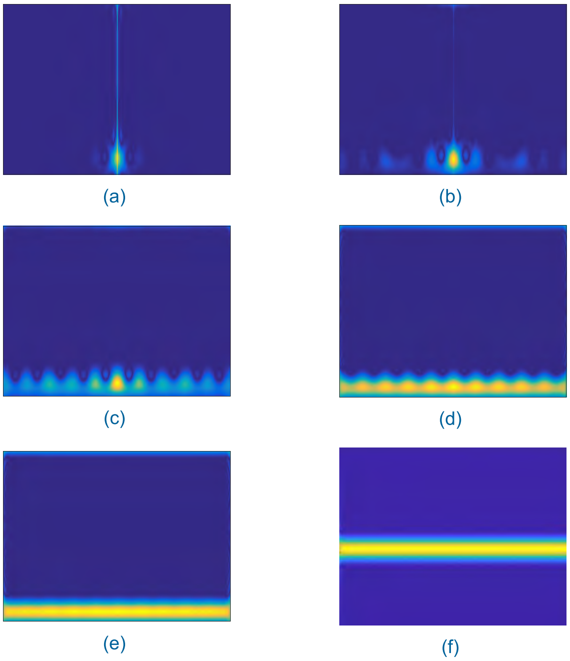

(a) (b) (c) (d) (e) (f)

FIGURE 8. The change of the feature image on the BPSK, the signal passes through five times autocorrelation. (a), (b), (c), (d), and (e) respectively represent the feature image of the BPSK obtained by one, two, three, four, and five times autocorrelation combined with the time-frequency analysis, and (f) is the TFI of the CP.

E. AUTOCORRELATION IMAGE ANALYSIS OF SIGNAL MULTIPLE AUTOCORRELATION

In order to get more features that can characterize the signal, we choose to construct the feature images by multiple autocorrelation. However, according to (15), (16), and (17), all autocorrelation values tend to be infinite after several iterative calculations of autocorrelation, so a reasonable number n of iterations should be chosen.

Consider the change in the signal autocorrelation image, when the SNR is 0dB, Fig. 8 shows the feature images of the BPSK obtained by autocorrelation, and the signal has undergone five times autocorrelation in total. We find that when $n \leq 4$, the feature images can represent the target signal. When $n > 4$, comparing Fig. 8(e) and Fig. 8(f), the Fig. 8(e) after 5 times autocorrelation is very close to the TFI of the CP in Fig. 8(f). In fact, Fig. 8(e) reflects the autocorrelation image formed when the autocorrelation values tend to infinity, and the changes caused by multiple autocorrelation always exist in the autocorrelation images. For example, the Fig. 9 shows the features images of multiple autocorrelation for the NCPM. It can be found that the feature image of the NCPM after several times autocorrelation are also similar to the CP. Therefore, the signal feature images formed by multiple autocorrelation are liable to be confused because they are similar to the TFI of the CP.

In order to reduce the effect of image distortion caused by multiple autocorrelation, We determine $n = 4$, i.e. four feature images are obtained through four times autocorrelation

$$\boldsymbol{trf}_n = CWD(y_n'(k)) \tag{19}$$

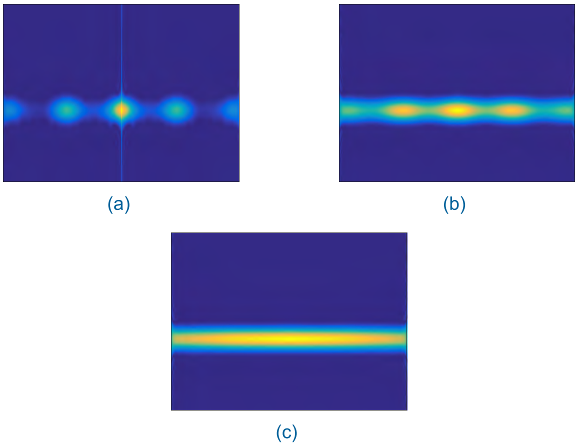

(a)
(b)
(c)

FIGURE 9. The change of the feature image on the NCPM after multiple autocorrelation.

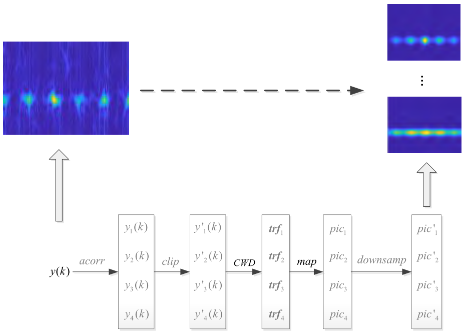


FIGURE 10. The pre-processing block diagram of multiple autocorrelation.

where $\boldsymbol{trf}_n$ denotes the two-dimensional matrix obtained by time-frequency transform after n times autocorrelation, and $n = 1, 2, 3, 4$. Finally, we substitute $\boldsymbol{trf}_n$ into (10) to obtain four feature images pic_n which can represent the same signal. It should be emphasized that we will make a detailed analysis in the final model analysis on the reasonable number of feature images in this paper.

In order to reduce the computational load of CPU and GPU, the obtained feature image is down sampled to image pic'_n with size of 64×64. We use $downsamp(\cdot)$ to denote the down sampling function. Fig. 10 shows the whole block diagram of the signal autocorrelation processing in this paper. After multiple autocorrelation processing, four feature images with size of 64×64 is finally obtained.

IV. PROPOSE A NOVEL CNN-BASED MODEL

In this section, we design a multiple autocorrelation joint decision (MAJD) model to fully extract the feature information of the four images we obtained in section III. The MAJD needs to input multiple images, so the method is suitable for the signal pre-processing method of this paper, which can obtain the pre-classification results of four feature images in the feature extraction stage. After that, an inference machine module based on fully connected structure is designed in MAJD, which combines pre-classification results to obtain better signal classification results.

A. DESIGN OF FEATURE EXTRACTION STRUCTURE

In [4]–[6] and [11], the initial input of the network is a TFI. In this paper, we propose to input four feature images that can represent the same signal. In order to make full use of the feature information of the four images, we construct a network which includes four groups of the CNN. As shown in Fig. 11, the feature image pic'_n is input into CNN n to complete training, and $n = 1, 2, 3, 4$. Each group of the CNN is completely consistent and the feature extraction stages are not connected with each other. They are only connected together in the inference machine.

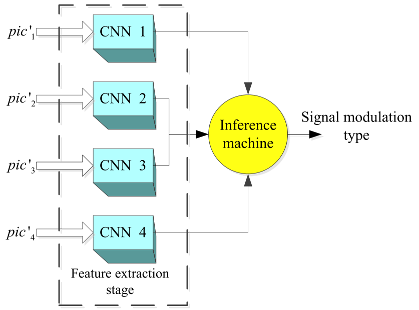


FIGURE 11. The overall structure diagram of the MAJD.

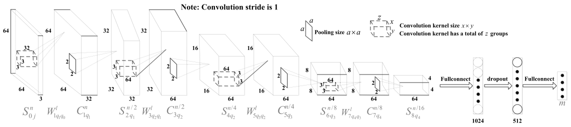


FIGURE 12. Unified structure used in four groups of the CNN.

Fig. 12 is a network structure uniformly adopted in the four CNN. We use m to represent the number of signal classes given by the classification task, and the CNNs' stride is 1. It consists of four convolution layers, four pooling layers, two fully connected layers and a dropout layer. The first layer of the convolution layers is input layer, and the second layer of the fully connected layers is the output layer. The dropout layer is designed to prevent over-fitting of the network. It temporarily discards neurons from the network according to a certain probability. In this paper, we set the probability value to 0.5. In Fig. 12, we need to emphasize that the final output of the fully connected layer is the result of pre-classification. Since the model is composed of four groups of the CNN, $4m$ pre-classification results will be obtained. Then, $4m$ pre-classification results are input into the inference machine to get the final recognition result. The structure of the inference machine will be described in detail in section IV-B.

According to the representation of the network model constructed in Fig. 12 and section II-B, the model uniformly adopted within the four CNNs can be expressed as follows

$$\begin{cases} C^{64}_{1q_1} = \sum_{q_0=1}^{3} (S^{64}_{0q_0} \otimes W^{l}_{1q_1q_0}) + b_{q_1} \\ \quad S^{32}_{2q_1} = Pooling(C^{64}_{1q_1}) q_1 = 1, 2, \ldots, 32 \\ C^{32}_{3q_2} = \sum_{q_1=1}^{32} (S^{32}_{2q_1} \otimes W^{l}_{3q_2q_1}) + b_{q_2} \\ \quad S^{16}_{4q_2} = Pooling(C^{32}_{3q_2}) q_2 = 1, 2, \ldots, 64 \\ C^{16}_{5q_3} = \sum_{q_2=1}^{64} (S^{16}_{4q_2} \otimes W^{l}_{5q_3q_2}) + b_{q_3} \\ \quad S^{8}_{6q_3} = Pooling(C^{16}_{5q_3}) q_3 = 1, 2, \ldots, 64 \\ C^{8}_{7q_4} = \sum_{q_3=1}^{64} (S^{8}_{6q_3} \otimes W^{l}_{7q_4q_3}) + b_{q_4} \\ \quad S^{4}_{8q_4} = Pooling(C^{8}_{7q_4}) q_4 = 1, 2, \ldots, 64 \\ [y_1, y_2, \ldots, y_{m-1}, y_m] = Fullconnect(S^{4}_{8q_4}) \end{cases} \tag{20}$$

B. DESIGN OF INFERENCE MACHINE STRUCTURE

The inference machine designed in this paper aims to make full use of the pre-classification results obtained in the feature extraction stage to make joint decision on the target class of the signal. The inference machine characterizes the probability that a pre-classification result may be the target class through the connection weights among neurons, which is equivalent to a fully connected network.

After the feature images are input into the network, we will obtain $4m$ pre-classification results. Since there is no training label in the pre-classification stage, the model pre-classifies a feature image into one of the classes. As shown in Fig. 6, we denote the result of pre-classification as p_{ij}, it denotes the probability that the i-th feature image is classified into class j, where i is the number of the CNN, $i = 1, 2, 3, 4$, j is the pre-classification result, $j = 1, 2, \ldots, m$, so p_{ij} has $4m$ values. The inference machine classifies again based on the results of pre-classification, and the number of classification classes is also m. The classification will finally determine the possible target class of the signal. Therefore, the training label is given in the training stage of the final classification. Given the pre-classification result p_{ij}, w_{ijk} represents the weight between the pre-classification result p_{ij} and the target class k, which can be expressed as

$$P(y = k|p_{ij}) = w_{ijk} \tag{21}$$

where y denotes the predictive class of the signal, so the actual meaning of the weight w_{ijk} denotes the probability that the signal may be the target class k when the feature image i is pre-classified as class j.

Finally, the inference engine determines the probability that the signal may be the target class k based on the pre-classification results

$$P(y = k) = \sum_{i} \sum_{j} p_{ij} w_{ijk} \tag{22}$$

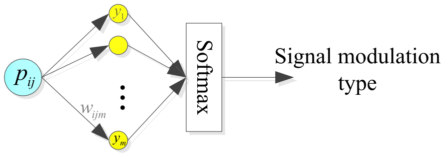


FIGURE 13. The structure of the inference machine.

We use the Softmax function to classify. For data sets that can be divided into m classes, $l \in \{1, 2, .., m\}$, the output probability can be express as

$$P\left(y = k\right) = \frac{\sum_{i} \sum_{j} p_{ij} w_{ijk}}{\sum_{l=1}^{m} \sum_{i} \sum_{j} p_{ij} w_{ijl}} \tag{23}$$

After classifying by Softmax function, the most probable class will output the maximum probability value, so the recognition result of the signal can be obtained by $\arg\max_{k \in \{1,2,\ldots,m\}} \left[P\left(y = k\right)\right]$.

C. NETWORK PARAMETER ADJUSTMENT

There are some parameters of network affecting the waveform recognition rate, such as the learning rate, the convolution kernel size, and the block size. Since the four CNNs are completely consistent in structure, we research the parameter configuration scheme of the first CNN, and the other three CNNs only need to copy the first one.

We try to find the optimal parameters structure by changing a certain variable and fixing the remaining variables. The performance of the network parameters is shown in Table 2. It should be emphasized that the test is carried out on the test set with the SNR of 9dB, and the training time consumption is based on the CPU of i5-3337U. And the simulation results are obtained after 1000 iterations.

TABLE 2. Hyper parametric tuning of the CNN model.

No.	Block size	Kernel size	Study rate	time assumption (min)	Recognition rate (%)
1	64	3×3	0.001	18	37.6
2	128	3×3	0.001	33	51.3
3	**256**	**3×3**	**0.001**	**48**	**65.5**
4	256	5×5	0.001	353	61.3
5	**256**	**7×7**	**0.001**	**674**	**65.8**
6	**256**	**3×3**	**0.01**	**50**	**71.7**
7	256	3×3	0.1	48	60.9
8	256	3×3	1	50	63.8

It can be found that when the block size is 256 (No. 3), the model can achieve a recognition rate of 65.5%. When we only change the size of the convolution kernel, it takes 674 minutes to iterate 1000 times on a 7×7 convolution kernel (No. 5). Although its recognition rate increases compared with No. 3, the time consumption cannot be ignored. So the configuration scheme of No. 5 is discarded. In addition, when the learning rate is 0.01, the recognition effect of the network is further improved (No. 6). Therefore, the parameter configuration scheme of the network model is determined as follows: block size is 256, convolution kernel size is 3 × 3, and learning rate is 0.01.

D. FINE TUNING NETWORK

Before the inference machine was trained well, the loss values were calculated between the m prediction results of $y_1, y_2, \ldots, y_m$ and the correct results (i.e., label values) $y_{1\_label}, y_{2\_label}, \ldots, y_{m\_label}$

$$loss = \frac{1}{m}\sum_{i=1}^{m}\left|y_i - y_{i\_label}\right|^2 \tag{24}$$

According to the loss value, the back-propagation algorithm [22] is used to fine tune the weight values in each convolution kernel and the fully connected layer, so that the internal weight parameters of the network are consistent with the types of the signals that need to be classified.

V. MODEL SIMULATION AND ANALYSIS

In this section, the paper evaluates the recognition performance of the MAJD model on the test set. Firstly, this paper gives the data set generation method of the simulation experiment. Secondly, the proposed MAJD model is compared with other radar signal waveform recognition methods. Finally, this paper makes a further structural analysis of the recognition performance of the MAJD.

A. THE SIMULATION ENVIRONMENT AND DATASET GENERATION

In reality environment, the carrier frequency and signal parameters of various types of signals keep changing at all time. To keep the simulation consistent with the actual electromagnetic environment, the values of the parameters of the simulated signals are randomly transformed within the specified range. Given the range (a, b), $U(a, b)$ denotes that the corresponding parameter values are uniformly distributed in the range a to b. Combing the parameters given in Table 1, the parameter range of the simulation signal is shown in Table 3, the signal sampling rate $f_s = 50$MHz. In Table 3, f_c, f_1, f_2, f_3 and f_4 denote the signal carrier frequency, τ_{pw} is pulse width, f_k is modulation frequency of the NCPM, L_c is code length of the BPSK, N is the number of sampling points on a coded value, and L is the code length of BFSK and QFSK. It should be emphasized that we generate BFSK and QFSK coding sequences by random numbers, and set corresponding thresholds for frequency switching.

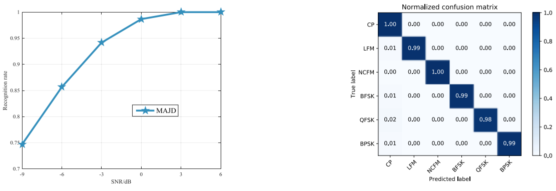
According to the parameter configuration of the six types of signals, the training set and test set are generated. The simulation generates data from a SNR of −9dB to 6dB in 3dB steps, and each SNR produces 800 samples for each type of signal. For training set and test set, we divide 800 samples generated by each type of signal into training set and test set according to the ratio of 3:1, so the training set consists of 21600 samples and the test set consists of 7200 samples. The model is built using Python's Tensorflow framework, and the GPU is Nvidia 1050Ti.

TABLE 3. The range of signal parameter values.

Modulation type	Parameters	Range
CP	f_c	$U(f_s/6, f_s/4)$
	τ_{pw}	$U(5e^{-6}, 20e^{-6})$
LFM	f_c	$U(f_s/6, f_s/4)$
	B	$U(f_s/20, f_s/10)$
	τ_{pw}	$U(5e^{-6}, 7.5e^{-6})$
NCPM	f_c	$U(f_s/6, f_s/4)$
	τ_{pw}	$U(5e^{-6}, 20e^{-6})$
	f_k	$U(f_s/100, f_s/50)$
BPSK	f_c	$U(f_s/6, f_s/4)$
	L_c	$\{7,11,13\}$
	N	$U(18,25)$
BFSK	f_1	$U(f_s/6, f_s/3)$
	f_2	$U(f_s/3, f_s/2)$
	L	$U(10,20)$
	N	$U(18,25)$
QFSK	f_1	$U(f_s/6, f_s/4)$
	f_2	$U(f_s/4, f_s/3)$
	f_3	$U(f_s/3, 5f_s/12)$
	f_4	$U(5f_s/12, f_s/2)$
	L	$U(15,25)$
	N	$U(18,25)$

B. THE RECOGNITION RESULT

As shown in Fig. 14(a), we acquire the overall recognition rate on the test set, it shows that with the improvement of the SNR, the recognition rate of the proposed model increases gradually. When SNR is −9dB, the recognition rate is 74%, and when the SNR is −6dB and −3dB, the recognition rate is 86% and 94% respectively. When SNR rises from −3dB to 6dB, the recognition rate of the MAJD can achieve more



(a) (b)

FIGURE 14. (a) The recognition result of the MAJD in the test set. (b) Normalized confusion matrix of the MAJD when the SNR is 0dB.

than 95% on the test set. At the same time, when the signal is 0dB, we give its confusion matrix as shown in Fig. 14(b). It can be found that the feature images of signals are easy to be confused with the feature image of the CP, which shows that the autocorrelation feature images designed in this paper still has some similarity, and the similarity has some influence on the classification of signal waveforms. In view of the similarity of images, we give a further analysis of this problem in section V-G.

C. COMPARISON TO CONVENTIONAL ALGORITHMS

In this paper, four conventional algorithms are selected to compare with the MAJD model proposed in this paper. In [24], it uses sparse classification (SC) to implement RSWR, and compares with the KNN algorithm. In [25], radial basis function support vector machine (RBF-SVM) is proposed to classify signals. In [26], stacked auto-encoders (SAE) are proposed to achieve RSWR, and the recognition results are compared with KNN, SC and RBF-SVM algorithm. Based on the simulation conditions of [26], we sets the same test samples and gets the recognition results under different SNR as shown in Fig. 14. It can be found that MAJD has better adaptability than conventional algorithms at low SNR. Specifically, when the SNR is −6dB, the recognition rate of MAJD is 89%, while the recognition rate of other algorithms is not more than 75%. With the improvement of SNR, when the SNR is greater than 0dB, the recognition rate of MAJD can reach almost 100%, while the recognition rate of other algorithms is still improving. Only when the SNR reaches 6dB, the recognition performance of SC and SAE is equal to MAJD.

D. COMPARISON TO THE DEEP LEARNING METHOD

We compare the recognition performance with the structure proposed in [4] and [6] using the deep learning method. In [6], the SCDAEs is proposed to reconstruct the original data, and then input the reconstructed data into the recognition network to achieve RSWR. It is noted that it also uses six types of signals for testing and the signal type is the same as us, so we don't change the signal type. When comparing with [6], we set the same simulation environment, therefore we produces test data with a SNR ranging from −8dB to 12dB, and each test signal produces 200 test samples per SNR. Fig. 14 shows a comparison of the recognition results of this paper with [6] on the test set.

In comparison with the recognition rate of [6], we finds that the MAJD has better recognition performance than the SCDAEs at low SNR environment. The main performance is that when the SNR is −8dB, the recognition rate of the MAJD for six types of signals can reach 75%, while that of the SCDAEs is 55%. When the SNR is increased from −8dB to −4dB, the MAJD and SCDAEs have a significant improvement in the waveform recognition rate. However, the degree of improvement of the SCDAEs is larger than that of the MAJD, which indicates that the SCDAEs is more sensitive to the noise effect of low SNR. In addition, when

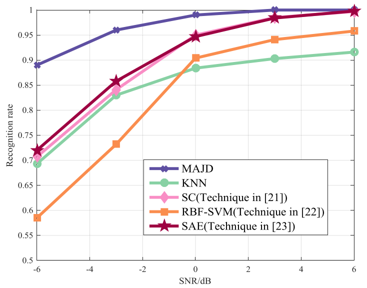
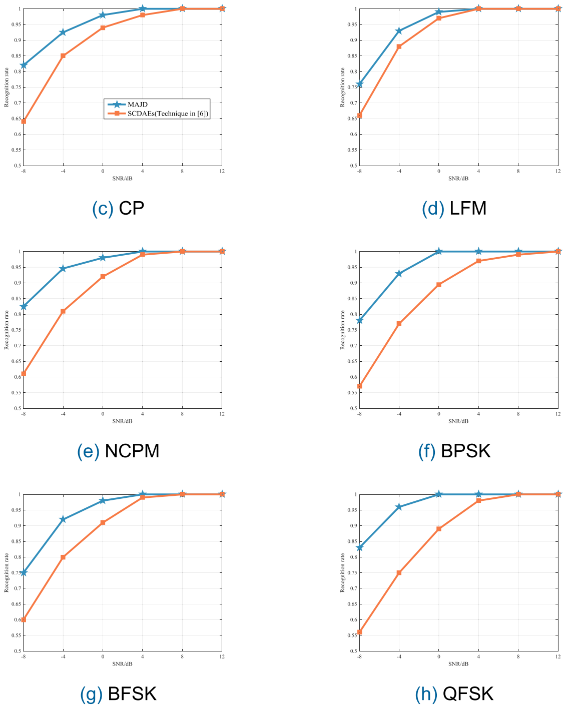


FIGURE 15. The recognition rate of five algorithms.



(c) CP

(d) LFM

(e) NCPM

(f) BPSK

(g) BFSK

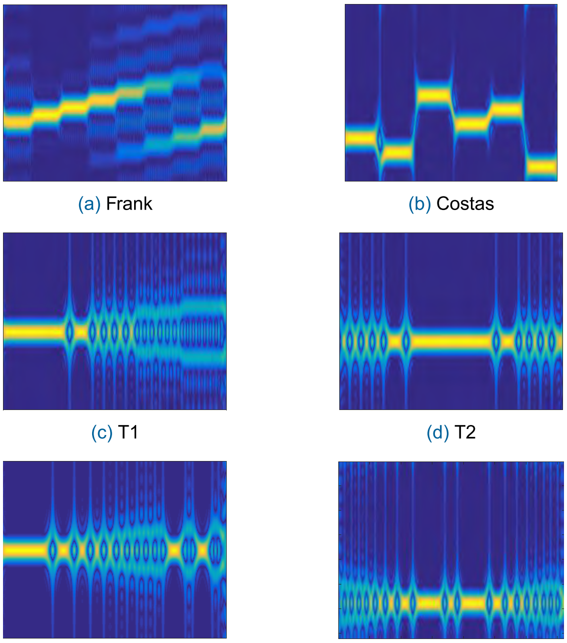
(h) QFSK

FIGURE 16. Comparison with the RSWR in [6].

the SNR is above 0dB, the recognition rate of the MAJD is above 97%. Compared with the SCDAEs, the recognition rate of the SCDAEs is above 97% only when the SNR is above 4dB. When the SNR is greater than 4dB, the recognition rate of the MAJD reaches 100%. The recognition performance of the SCDAEs is similar to that of the MAJD only when SNR is greater than 8dB.

In [4], the combination of image morphology and threshold filtering is proposed. After transforming the signal into a TFI, the threshold filtering is used to remove a large amount of noise of the TFI, and then the image morphology is used to remove the fine noise in the TFI. The paper simulates eight types of signals, including BPSK, LFM, Costas, Frank, and T1-T4 codes. Since the Costas, Frank and T1-T4 codes are not given in the simulation of our paper, the six types of the signals are first simulated in this paper. Their TFIs are shown in Fig. 17.

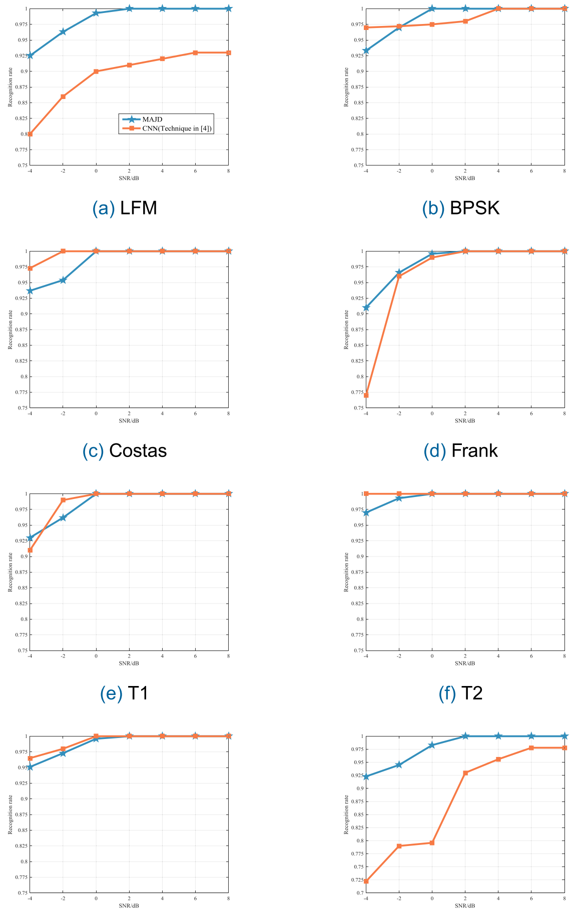
Then we use the simulation conditions of [4], change the range of parameters of eight types of signals and set up


(a) Frank (b) Costas

(c) T1 (d) T2

(e) T3 (f) T4

FIGURE 17. The time-frequency images of the signal in [4].



(a) LFM (b) BPSK

(c) Costas (d) Frank

(e) T1 (f) T2

(g) T3 (h) T4

FIGURE 18. Comparison with the RSWR in [4].

corresponding test sets. Therefore, this paper produces test data with a SNR ranging from −4dB to 8dB, and each test signal produces 1000 test samples per SNR. Fig. 18 shows the comparison of the recognition results of the two models on the test set.

In comparison with the recognition rate of [4], we find that the MAJD has better comprehensive recognition performance than the algorithm of [4]. It is mainly manifested in two aspects: recognition rate and recognition stability. From the perspective of recognition rate, when SNR is −4dB, the recognition rate of BPSK, Costas, T2 and T3 of [4] is higher than that of the MAJD. The recognition rate of the four signals is over 96% while that of the MAJD is 93%. For LFM, Frank, T1 and T4, the recognition rate of the MAJD is higher than that of [4], the MAJD is more than 91% while the recognition rate in [4] is more than 72%. Therefore, the MAJD achieves a better result in terms of overall recognition rate.

From the perspective of recognition stability, the trend of recognition rate in [4] is difficult to find, and the recognition rate of signals varies greatly. The recognition rates of BPSK, Costas, T1, T2 and T3 are higher than LFM, Frank and T4. In LFM, Frank, and T4, the LFM curve changes gently, while the curves of Frank and T4 change steeply. In contrast, MAJD has a relatively consistent trend in the curve of eight types of signals, so it has higher stability.

E. ANALYSIS TO MAJD MODEL: THE EFFECT OF THE NUMBER OF FEATURE IMAGES ON THE CLASSIFICATION RESULTS

Based on the design idea of MAJD, we infer that the number of input feature images has a great impact on the recognition rate of the network, so we further analyze the structure of the designed network.

In MAJD, we analyze it by changing the number of input feature images and the network structure. When we change the number of input feature images, the number of CNNs changes accordingly. For example, when we input three feature images, the fourth group CNN in Fig. 11 will be discarded. Fig. 19 shows the effect of the number of input feature images on the waveform recognition rate, and we compares it with the overall recognition rate of SCDAEs. It should be emphasized that the training set and test set are generated in the same way as section V-A, so the training set and the test set have 21600 and 7200 samples respectively.

In the simulation, it is found that the number of input feature images has a great influence on the recognition rate. When $n = 1$ and the SNR is below −8dB, the recognition rate of MAJD is equivalent to that of SCDAEs. At this time, the proposed algorithm does not generate multiple feature images for joint decision-making. When $n > 1$, the recognition rate is higher than that of SCDAEs, so it can be known that the increase of the number of feature images plays an important role in waveform recognition at low SNR. It is

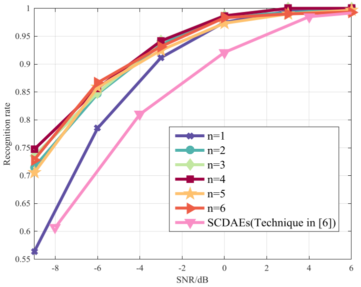


FIGURE 19. The effect of the number of input feature images.

TABLE 4. The effect of channel number.

Number of channels	-9dB	-6dB	-3dB	0dB	3dB	6dB
1	0.563	**0.785**	**0.912**	**0.975**	**0.995**	**1**
2	**0.623**	0.745	0.87	0.952	0.983	1
3	0.583	0.743	0.855	0.903	0.965	0.98
4	0.538	0.725	0.772	0.855	0.927	0.933
5	0.562	0.677	0.758	0.84	0.905	0.945
6	0.563	0.675	0.772	0.807	0.872	0.87

TABLE 5. The effect of the number of CNN groups.

Number of CNN groups	-9dB	-6dB	-3dB	0dB	3dB	6dB
1	0.563	0.785	0.912	0.975	0.995	1
2	0.715	0.847	0.935	0.987	0.992	1
3	0.735	0.848	0.937	0.983	1	1
4	**0.747**	0.857	**0.942**	**0.987**	**1**	**1**
5	0.705	0.858	0.925	0.973	0.99	0.997
6	0.728	**0.867**	0.932	0.983	0.99	0.993

noted that as the number of input feature image increases, the recognition rate of MAJD does not gradually increase. When $n = 2$, the recognition rate of the model is 15% higher than that of $n = 1$, but the improvement of the recognition rate after $n \geq 2$ is not very obvious. Specifically, when the SNR is -9dB, the number of feature images of $n \geq 2$ leads to a recognition rate between 70% and 75%. When $n = 2$, the recognition rate is 71%, when $n = 3$, the recognition rate is 73%, when $n = 4$, the recognition rate is 74%, and when $n = 5$ and $n = 6$, the recognition rate is 70% and 73% respectively. We found that when $n \leq 4$, the recognition rate of the model still improved, but when $n > 4$, the recognition rate begins to decline. In fact, the fluctuation of recognition rate in MAJD is reasonable. According to section III-D, when $n \leq 4$, we can effectively characterize signals through images generated by multiple autocorrelation. But when $n > 4$, the feature images generated by autocorrelation are difficult to classify because they are very similar, which is the reason that the recognition rate begins to decline. Therefore, it further illustrates that the number of four feature images given in this paper is reasonable. Through the analysis at low SNR, we can know that increasing the number of autocorrelation feature images can indeed improve the waveform recognition rate, but we cannot obtain the feature images indefinitely through autocorrelation iteration calculation, a reasonable value should be $n \leq 4$. But if we consider the factors such as recognition rate and computational difficulty, a better number of feature images should be $n = 2$, because the recognition rate is much higher than $n = 1$, while the improvement of recognition rate of $n = 3$ and $n = 4$ is smaller.

At high SNR, increasing the number of feature images does not significantly improve the recognition performance. When SNR is greater than 0dB, no matter how many feature images are input, the network structure of this paper can achieve more than 97% recognition rate.

F. ANALYSIS TO MAJD MODEL: THE EFFECT OF THE FEATURE IMAGE INPUT MODE ON CLASSIFICATION RESULTS AND TIME CONSUMPTION

The MAJD proposed in this paper essentially uses a CNN for feature extraction for each feature image, but for the structure of CNN, it has its own way of extracting multiple feature images, i.e. its multiple channel. As we mentioned in section II-B, since the TFI is usually gray-scale image, the number of channels of CNN is set to 1. However, for this paper, we get several feature images by autocorrelation, which means that they can be input into different channels of CNN, so the CNN can also extract feature of each image. Based on this consideration, we further analyze the effect of input mode on waveform classification. In this section, the training set and test set are generated in the same way as section V-A.

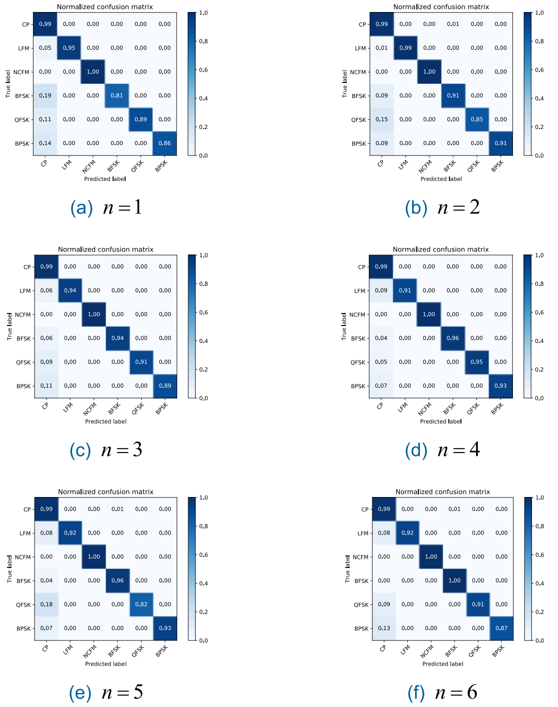
Table 4 shows the waveform recognition results of the CNN structure of Fig. 12 under different channels, and Table 5 shows the waveform recognition results of the different CNN groups of the MAJD. In Table 4, we found that inputting multiple feature images to different channels cannot improve the waveform recognition results, and even lead to a decrease in recognition results. Specifically, when $SNR \geq -6$dB, the recognition rate of the CNN is the highest when the number of channels is 1. With the increase of the number of channels, the recognition result shows a downward trend. Finally, when $SNR = 6$dB and the number of channel is 6, the recognition result is only 87%, which is a significant decrease compares with the 100% recognition rate obtained when the channel number is 1. While in Table 5, the structure of the MAJD is more efficient for classification based

TABLE 6. Time consumption of different feature images.

Number of feature images	Data preprocessing	Feature extraction and classification (channel/group)
1	0.026s	0.0015s/0.0015s
2	0.053s	0.0016s/0.0026s
3	0.081s	0.0017s/0.0038s
4	0.105s	0.0017s/0.0050s
5	0.138s	0.0019s/0.0063s
6	0.160s	0.0021s/0.0078s

on multiple feature images. Specifically, when the SNR is −9dB and the number of feature images is greater than 2, the recognition rate is greater than 70%, which is at least 10% higher than that of feature extracted by channel. Compared with extracting image features in channel mode, with the increase of the number of CNN groups, the recognition rate of MAJD in high SNR increases steadily. And no matter how many feature images are input, the recognition performance of MAJD is better than that in channel mode. According to the comparison of the recognition results in Table 4 and Table 5, the following conclusions can be drawn: MAJD can make full use of multiple feature images to improve the recognition results effectively. However, the multi-channel input method cannot make full use of the effective pixel information generated between each feature image, which will lead to the decline of recognition results based on the feature images designed in this paper. Therefore, if we need to extract the feature image information in this paper, the better way is to use multiple parallel CNN networks, rather than changing the number of input channels of the network.

However, although several parallel CNNs improve the waveform recognition rate, they also lead to more time consumption, as shown in Table 6. In Table 6, the average time consumption of single sample is calculated by Monte Carlo simulation. The number of simulation times is 500, the CPU is i5-7300HQ and the GPU is GTX 1050Ti. It can be found that with the increase of the number of feature images, the time consumption of a sample increases gradually in the data preprocessing stage. When the number of feature images is 6, the data preprocessing stage will consume 0.16s, which increases the time consumption by 6 times compared with 1 feature image. In the stage of feature extraction and classification, it does not consume much time to extract feature information from multiple images by channel mode, but extracting feature information from multiple parallel CNNs results in a gradual increase in time consumption. The average time consumption reveals that the MAJD suffer more computation burden than the conventional CNN structure, the increase of the number of CNN groups will reduce the recognition speed of the model.



(a) $n = 1$ (b) $n = 2$ (c) $n = 3$ (d) $n = 4$ (e) $n = 5$ (f) $n = 6$

FIGURE 20. The change in the confusion matrix when the number of feature images changes. The SNR is −3dB.

G. ANALYSIS TO MAJD MODEL: ANALYSIS OF THE EFFECT OF MULTIPLE FEATURE IMAGES ON IMAGE SIMILARITY

In Section III-D, aiming at the similarity of feature images, this paper proposes that more features of signals can be extracted from multiple autocorrelation images to overcome the bad influence of similarity on waveform classification results. In view of this problem, we will make further analysis in this section.

When SNR is −3dB, we give the confusion matrix obtained at different number of feature images as shown in Fig. 20. It should be emphasized that due to the problem of display accuracy, a few values less than 0.01 in the confusion matrix are displayed to be 0, so the sum of some signal classification values is not 1 but 0.99. In Fig. 20, we find that the autocorrelation feature images of various signals are easily confused with the autocorrelation feature image of CP. However, the degree of confusion can be reduced by extracting features from multiple feature images. When the number of input feature image is 1, the misclassification values of LFM, BFSK, QFSK, and BPSK to CP are 0.05, 0.19, 0.11, and 0.14, respectively. As the number of feature image increases, the misclassification value is further reduced. When $n = 4$, the overall classification result is optimal, and the misclassification values of LFM, BFSK, QFSK and BPSK to CP are 0.09, 0.04, 0.05 and 0.07, respectively. However, when the number of input feature image continues to increase, the misclassification results cannot be further reduced, and the trend of change is more uncertain (see Fig. 20(e) and Fig. 20(f)). In fact, when $n > 4$, the feature

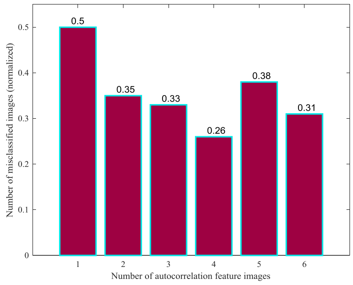


FIGURE 21. The total number of misclassified images when the number of feature images changes, and the value is normalized. The SNR is −3dB.

images obtained by the five and six times autocorrelation becomes similar, which increases the difficulty of waveform classification and makes the signals easily confused. We calculate all the misclassification results under the different number of feature images. And the overall change of this misclassification result is more obvious in Fig. 21. We find that when $n \leq 4$, the misclassification results gradually decrease, but when $n > 4$, the performance of the model is not as good as $n = 4$. Therefore, like section V-E,an important conclusion is that $n = 4$ is the maximum number of feature images, if we want to overcome the effect of image similarity through multiple autocorrelation feature images, it is better to control n below 4, because only in this way, the feature images of each signal are not easily confused, and the pixel characteristics extracted by the MAJD can effectively represent the target signal.

VI. CONCLUSION

The MAJD proposed in this paper provides more initial information for the network by inputting multiple feature images. Each feature image is classified by the CNN and achieve a result by the inference machine. In the multiple autocorrelation, in order to prevent the distortion of TFIs of the signals, we choose the number of autocorrelation to be 4, so we can obtain four TFIs which can represent the same signal. For the parameter configuration of the MAJD, this paper presents a hyper parametric optimization scheme of the MAJD and applies it to each group of the CNN. In the testing phase, the MAJD is tested by two groups of test sets, which contain six and eight types of the signals respectively. Finally, the influence of the MAJD on the waveform recognition rate is further analyzed. The simulation results compared with the literature show that the proposed MAJD model not only has high recognition rate at high SNR, but also is more suitable for waveform recognition at low SNR environment. In the final comprehensive analysis of the MAJD, we analyzes the influence of the number of autocorrelation feature images on the recognition rate, time consumption, and image similarity. The analysis verifies the rationality of 4 as the number of feature images, but $n = 2$ may be a better choice considering the computational time consumption. At the same time, comparing the channel input mode verifies the validity of the proposed network structure for extracting multiple feature images. From the perspective of recognition rate and recognition stability, the MAJD model achieves better recognition performance.

REFERENCES

[1] R. G. Wiley, *ELINT: The Interception and Analysis of Radar Signals*. Norwood, MA, USA: Artech House Press, 2006.

[2] E. R. Zilberman and P. E. Pace, "Autonomous time-frequency morphological feature extraction algorithm for LPI radar modulation classification," in *Proc. Int. Conf. Image Process.*, Oct. 2006, pp. 2321–2324.

[3] M. Zhang, L. Liu, and M. Diao, "LPI radar waveform recognition based on time-frequency distribution," *Sensors*, vol. 16, no. 10, p. 1682, Jul. 2016.

[4] M. Zhang, M. Diao, and L. Guo, "Convolutional neural networks for automatic cognitive radio waveform recognition," *IEEE Access*, vol. 5, pp. 11074–11082, 2017.

[5] S.-H. Kong, M. Kim, L. M. Hoang, and E. Kim, "Automatic LPI radar waveform recognition using CNN," *IEEE Access*, vol. 6, pp. 4207–4219, 2018.

[6] Z. Zhou, G. Huang, H. Chen, and J. Gao, "Automatic radar waveform recognition based on deep convolutional denoising auto-encoders," *Circuits, Syst., Signal Process.*, vol. 37, no. 9, pp. 4034–4048, Jan. 2018.

[7] G. López-Risueño, J. Grajal, and A. Sanz-Osorio, "Digital channelized receiver based on time-frequency analysis for signal interception," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 3, pp. 879–898, Jul. 2005.

[8] T. Chen, L. Liu, and X. Huang, "LPI radar waveform recognition based on multi-branch MWC compressed sampling receiver," *IEEE Access*, vol. 6, pp. 30342–30354, 2018.

[9] J. Zhu, Y. Zhao, and J. Tang, "Automatic recognition of radar signals based on time-frequency image character," in *Proc. IET Int. Radar Conf.*, Apr. 2013, pp. 1–6.

[10] C. Wang, J. Wang, and X. Zhang, "Automatic radar waveform recognition based on time-frequency analysis and convolutional neural network," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 2437–2441.

[11] Q. Wang, P. Du, J. Yang, G. Wang, J. Lei, and C. Hou, "Transferred deep learning based waveform recognition for cognitive passive radar," *Signal Process.*, vol. 155, pp. 259–267, Feb. 2019.

[12] X. Zhang, P. Luo, and X. Hu, "A hybrid method for classification and identification of emitter signals," in *Proc. 4th Int. Conf. Syst. Inform. (ICSAI)*, Nov. 2017, pp. 1060–1065.

[13] T. R. Kishore and K. D. Rao, "Automatic intrapulse modulation classification of advanced LPI radar waveforms," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 53, no. 2, pp. 901–914, Apr. 2017.

[14] M. Zhang, M. Diao, L. Gao, and L. Liu, "Neural networks for radar waveform recognition," *Symmetry*, vol. 9, p. 75, May 2017.

[15] L. Liu, S. Wang, and Z. Zhao, "Radar waveform recognition based on time-frequency analysis and artificial bee colony-support vector machine," *Electronics*, vol. 7, no. 5, p. 59, Mar. 2018.

[16] D. Zeng, X. Zeng, G. Lu, and B. Tang, "Automatic modulation classification of radar signals using the generalised time-frequency representation of Zhao, Atlas and Marks," *IET Radar, Sonar Navigat.*, vol. 5, no. 4, pp. 507–516, Apr. 2011.

[17] D. Shi and Y. Gao, "Electromagnetic radiation source identification based on spatial characteristics by using support vector machines," *Appl. Comput. Electromagn. Soc. J.*, vol. 32, no. 2, pp. 120–127, Feb. 2017.

[18] L. Gao, X. Zhang, J. Gao, and S. You, "Fusion image based radar signal feature extraction and modulation recognition," *IEEE Access*, vol. 7, pp. 13135–13148, 2019.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105.

[20] J. Lunden and V. Koivunen, "Automatic radar waveform recognition," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 1, pp. 124–136, Jun. 2007.

[21] J. Z. Gao. *Weak Signal Detection*, 2th ed. Beijing, China: Tsinghua Univ. Press, 2011.

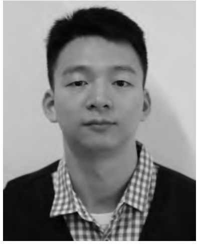
[22] G. Ian, B. Yoshua, and C. Aaron, *Deep Learning*. Cambridge, MA, USA: MIT Press. 2016.

[23] G. Hu, *Modern Signal Processing Tutorial*. Beijing, China: Tsinghua Univ. Press. 2004.

[24] J. Ma, G. Huang, W. Zuo, X. Wu, and J. Gao, ‘‘Robust radar waveform recognition algorithm based on random projections and sparse classification,’’ *IET Radar, Sonar Navigat.*, vol. 8, no. 4, pp. 290–296, 2014.

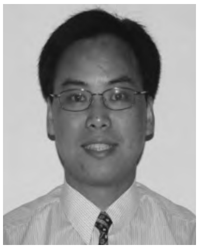
[25] Y. Shi, H. Ji, M. Zhu, and L. Wang, ‘‘Specific radar emitter identification in multiple kernel fusion framework,’’ *J. Electron. Inf. Technol.*, vol. 36, no. 10, pp. 2484–2490, 2014.

[26] Z. Zhou, G. Huang, J. Gao, and X. Man, ‘‘Radar emitter identification algorithm based on deep learning,’’ *J. Xidian Univ.*, vol. 44, no. 3, pp. 85–90, Jun. 2017.



ZHI HUANG was born in Fujian, China, in 1994. He received the B.S. degree in information countermeasure from the Naval University of Engineering, Wuhan, Hubei, in 2017, where he is currently pursuing the M.E. degree in circuits and systems. His research interests include emitter recognition, deep learning, and signal processing.

ZHIYUAN MA was born in Hubei, China, in 1982. He received the M.E. degree in electronic circuit and system from Central China Normal University, China, in 2007, where he is currently pursuing the Ph.D. degree in radio physics. He is currently an Assistant Professor with the Department of Electronic Technique, Naval University of Engineering, China. His research interest includes intelligent signal processing.



GAOMING HUANG was born in Hunan, China, in 1972. He received the B.S. and M.E. degrees in electronic warfare from the Naval Electronic College of Engineering, China, in 1995 and 1998, respectively, and the Ph.D. degree in signal processing from Dongnan University, Nanjing, China, in 2006. He is currently the Chief Professor with the Department of Information Countermeasure, Naval University of Engineering, China. His research interests include passive detection, intelligent signal processing, and electronic warfare system simulation.