IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 8, 2019, accepted October 10, 2019, date of publication October 18, 2019, date of current version December 11, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2947536

MRD-Nets: Multi-Scale Residual Networks With Dilated Convolutions for Classification and Clustering Analysis of Spacecraft Electrical Signal

YIXIN LIU[1], KE LI[1], (Member, IEEE), YUXIANG ZHANG[1], AND SHIMIN SONG[2]

[1]Fundamental Science on Ergonomics and Environment Control Laboratory, School of Aeronautic Science and Engineering, Beihang University, Beijing 100191, China

[2]China Academy of Space Technology, Beijing 100048, China

Corresponding author: Ke Li (like@buaa.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61773039, in part by the Aeronautical Science Foundation of China under Grant 2017ZDXX1043, and in part by the Aeronautical Science Foundation of China under Grant 2018XXX.

ABSTRACT The fault detection of spacecraft electronic load systems is a crucial part of the spacecraft prognostics and health management system. To detect the abnormal state of spacecraft electronic load systems, complex electrical signals should be processed rapidly and accurately. For the fault detection of spacecraft electronic load systems, a robust unsupervised clustering analysis method and an accurate supervised classification method are of great importance. However, the traditional machine learning methods have poor performance when processing high-dimensional signal data because of the lack of ability to extract complex features from the signals. Therefore, neural-network-based deep learning (DL) models which can extract features from signals automatically are more suitable in this situation. In this paper, a novel convolutional neural network (CNN) module, the multi-branch residual module with dilated convolutions (MRD module), is proposed to extract multi-scale features from the electrical signal. Then, a well-designed CNN model named MRD-CNN is presented for the supervised classification task of signal. Furthermore, for the unsupervised clustering task, the clustering variational autoencoder with MRD modules (MRD-CluVAE) is proposed. The MRD-CluVAE can extract high-quality features from signal data and output the clustering results directly. To evaluate the performance of the proposed models, comparisons among the proposed models and other baseline algorithms are carried out. The experimental results show that the MRD-CNN model achieves higher classification performance and stability, while the MRD-CluVAE has a higher clustering accuracy than other algorithms. These methods can be utilized to resolve the classification and recognition problems of spacecraft electronic load signals.

INDEX TERMS Fault detection, spacecraft electrical signal, MRD module, MRD-CNN, MRD-CluVAE.

I. INTRODUCTION

With the development of space technology, spacecrafts and their subsystems have become more automated and complex [1]. Once a fault occurs in a spacecraft or its subsystems, the system cannot function well. Sometimes, this will lead to the ineffectiveness of the entire spacecraft. Thus, detecting faults and predicting possible diagnostics for spacecraft become necessary. In the subsystems of spacecraft, the electronic load systems are typically nonlinear time-dependent subsystems. Thus, fault detection in these systems is a complicated problem [2]. Therefore, a rapid, accurate and robust fault detection method is needed for spacecraft electronic load systems.

To monitor the working position and detect abnormal operations in spacecraft electronic load systems in real time, complex time-dependent electrical signals should be analyzed and recognized rapidly and accurately. In previous studies,

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

we have assessed different machine learning methods for the fault detection of spacecraft electronic load systems, such as weighted proximal support vector machine (WPSVM) and random forest (RF) for online classification [2], [3], and fuzzy C-means (FCM) clustering for offline building of expert dataset [4].

However, these methods cannot perform well on the signal processing. On the one hand, for the online classification, since the long time-series signal is high-dimensional and noisy, feature extraction algorithms, such as principal component analysis (PCA) and deep belief network (DBN), are applied to reduce the dimensions of raw signal data, and then the low-dimensional feature vectors are fed into the classification algorithms (e.g. WPSVM and RF). In such methods that combine feature extraction algorithms with classifiers, there is a gap between these two parts, which means that the classification results cannot feedback to the process of feature extraction. Thus, the features extracted by such algorithms may contain redundant information, and some of the features are not representative for the specific task. On the other hand, an effective clustering algorithm is of great significance for data annotation when building the expert training dataset in the offline system. However, for the clustering task of high-dimensional signal data, classical clustering analysis methods, such as K-means and Density-Based Spatial Clustering of Application with Noise (DBSCAN), are ineffective, while the performance of FCM still can be improved. Therefore, classification algorithms as well as clustering algorithms with high performance are urgently needed for the processing of high-dimensional signals.

Recently, a new enthusiastic research field, deep learning (DL), makes many breakthroughs in image and speech recognition [5]. Commonly used DL methods include the multilayer perceptron (MLP), autoencoders (AE), convolutional neural networks (CNNs) and recurrent neural networks (RNNs) [6]. Compared with traditional machine learning algorithms, such DL methods possesses special advantages, such as the ability for automatic feature extraction and the flexibility of architectures.

In particular, CNNs, as one of the well-applied DL models, has become the dominant approach for almost all recognition and detection tasks in the research domain of computer vision and speech recognition [7]. In comparison with other algorithms, CNNs have excellent capabilities to learn features from raw image data. Moreover, due to the similarity between electrical signal recognition and the previously mentioned application areas, CNN methods ought to perform better than traditional methods in signal classification problems [8].

Moreover, in the field of unsupervised learning, AE is a successful model for dimension reduction and data generation [9]. By setting the target values of the neural networks to be equal to the inputs, AE can extract low-dimensional features from high-dimensional data in an unsupervised way [10]. Many AE-based approaches also achieve high performance on clustering analysis [11], which can be beneficial for the building of expert database of signals.

In this paper, for the fault detection of the spacecraft electronic load system, a framework consisting of online and offline systems is introduced. Then, a novel neural network module, the Multi-branch Residual module with Dilated convolutions (MRD module), is designed for the feature extraction of signal data. For the two critical steps in the framework of fault detection, we propose two DL models for the data processing respectively. Specifically, the convolutional neural network with MRD modules (MRD-CNN) model is proposed to classify the electrical signal in the online system. Moreover, the clustering variational autoencoder with MRD modules (MRD-CluVAE) model is built up for clustering analysis of the electrical signal in the offline system, which is beneficial for the building of the expert dataset. The main contributions of this article are the following aspects:

1. A processing pipeline of multichannel signals for spacecraft electronic load systems is suggested, which includes offline data pre-processing, clustering analysis, error correction and online classification for fault detection.
2. A novel one-dimensional convolutional module named MRD module is proposed to extract multi-scale features from the electrical signal. The MRD module can be flexibly applied in the classification model and clustering model for signal processing.
3. A DL model named MRD-CNN is proposed for the signal classification of spacecraft electronic load systems. The MRD module is applied to the feature extraction part of the model. Meanwhile, to enhance the classification performance, many advanced techniques for CNNs are applied.
4. A new clustering analysis model based on variational autoencoder (VAE) is put forward for the clustering of the electrical signal. On the basis of VAE, the model is improved to output clustering results directly, and utilizes the MRD module for the construction of VAE.
5. Experiments and analysis of the MRD-CNN and MRD-CluVAE models are carried out to evaluate and compare the proposed models with baseline methods. The results show that the models perform better than other approaches.

The structure of the remainder of this paper is organized as follows. Related works are given in Section II. In Section III, the methodology is introduced. In section IV, the results of experiments are discussed, including the results of the classification and clustering analysis. Finally, conclusions are drawn in Section V.

II. RELATED WORKS

In this section, existing works related to our research are briefly discussed.

A. CNNS FOR ONE-DIMENSIONAL DATA'S PROCESSING

Proposed by Lecun *et al.* [12], CNNs are one of the prevalent DL models recently. CNNs use convolutional layers and activation function to extract features from the input, while the pooling layers play a role in down-sampling to make the model extract higher-level features from a larger-scale

input [6]. Similar to other DL models, the most significant character of CNNs is the capability of extracting features automatically. Also, the architecture of CNNs is flexible, which means that it is possible to modify the architectures according to the needs of idiographic tasks and datasets. In addition, the working mechanism of convolutional filters contributes to CNNs' advantages in processing continuous structural data, such as images data and time-series sequence data. In recent years, CNNs have been hlwidely used in a variety of computer vision tasks, such as image classification, localization, detection, and segmentation [13]. Meanwhile, many advanced architectures of CNNs, such as ResNet [14] and GoogLeNet [15], are proposed for accurate classification of images.

Different from image processing tasks, when processing one-dimensional time-series data, the filters of the layers in CNNs should be adjusted to a one-dimensional structure. The CNNs employing one-dimensional filters are defined as 1D-CNNs. Similar to the two-dimensional convolutional layer, the output of the one-dimensional convolution layer can be calculated as

$$c_i^{l,j} = f\left(b_j^l + \sum_{m=1}^{M} w_m^{l,j} x_{i+m-1}^{l-1,j}\right), \tag{1}$$

where l is the layer index, f is the activation function, b_j is the bias term for the j^{th} feature map, M is the filter size, $c_i^{l,j}$ is the output of the convolutional layer for feature map j and sequence index i, $x_k^{l-1,j}$ is the output of the $(l-1)^{th}$ layer for feature map j and sequence index k, and w_m^j is the weight for feature map j and filter index m.

The output of one-dimensional max pooling layer is given as

$$p_i^{l,j} = \max_{r \in R}\left(x_{i\times T+r}^{l-1,j}\right), \tag{2}$$

where l is the layer index, R is the pooling size, $p_i^{l,j}$ is the output of the pooling layer for feature map j and sequence index i, and $x_k^{l-1,j}$ is the output of the $(l-1)^{th}$ layer for feature map j and sequence index k.

1D-CNNs have been applied in a variety of classification tasks. For example, Eren advanced a bearing fault detection method using 1D-CNNs [16], Ronao and Cho utilized deep 1D-CNNs for a human activity recognition task [5], Dan *et al.* suggested a real-time ECG classification method based on CNNs [17], and Jing *et al.* applied CNN to diagnose fault of gearbox [7]. Using advanced techniques, a novel 1D-CNN module is proposed for feature extraction of signals, which can be used in the models for signal classification and clustering analysis.

B. VAE FOR CLUSTERING ANALYSIS

Introduced by P and Max in [18],VAE is a successful model for unsupervised feature extraction and data generation [11]. Unlike the general AE, VAE combines variational Bayesian (VB) approach and neural networks which are used as probabilistic encoders and decoders [18]. It works efficiently when the posterior distribution is intractable and/or the dataset is large. In VAE, the input of the encoder network is denoted as x and the latent variable output by the encoder network is denoted as z. The conditional posterior distribution $p_\theta(z|x)$ is approximated by another distribution $q_\phi(z|x)$ [19]. As a result, the intractable inference problems can be converted into an optimization problem which can be solved via methods like stochastic gradient descent (SGD) and backpropagation through time (BPTT). According to the variational principle, a lower bound can be used as a loss function to optimize the parameters [20]. The lower bound of VAE is

$$L\left(\theta, \phi; x^{(i)}\right) = \mathrm{KL}\left(q_\phi\left(z|x^{(i)}\right)||p_\theta(z)\right) - E_{q_\phi(z|x^{(i)})}\left[\log p_\theta\left(x^{(i)}|z\right)\right], \tag{3}$$

where KL refers to Kullback-Leibler divergence. KL is used to estimate the difference between two probability distribution. For example,

$$\mathrm{KL}(p(x)||q(x)) = \int p(x)\log\frac{p(x)}{q(x)}dx. \tag{4}$$

For unsupervised clustering analysis tasks, the variants of VAE have been extensively studied. For example, Zhang *et al.* proposed variational deep embedding (VaDE), which is an unsupervised and generative clustering framework [11]. Dilokthanakul *et al.* introduced a deep unsupervised clustering method, Gaussian mixture variational autoencoder (GMVAE), which combines Gaussian mixture with VAE [9]. Similar to GMVAE, in [21], Nalisnick *et al.* presented the deep latent Gaussian mixture model (DLGMM) that takes advantage of VAE and Gaussian mixture models (GMM). Different from these works, we modify the network structure and loss function of VAE to output clustering results directly. At the same time, our approach uses a novel convolution module to replace the fully connected layer in VAE to enhance the performance of feature extraction and reconstruction ability of the model.

III. METHODOLOGY

In this paper, the multimodal signal data for the fault detection of spacecraft electronic load systems is acquired from a multichannel spacecraft measurement system. Specifically, a total of 50 channels of data are detected. The flow chart of the fault detection for spacecraft electronic load systems is shown in Fig. 1.

The process is divided into two parts, the offline part, and the online identification part. In the offline system, the historical data need to be pre-processed, including segmentation and wavelet denoising. Then, the unlabeled signal data should be categorized into different sample types for the training of the classification algorithm. Instead of time-consuming manual labeling, a clustering-assisted annotation is adopted to categorize the unlabeled electrical signal data. In the first step of the clustering-assisted annotation, data is fed into the clustering analysis algorithm to obtain temporary labels. Then, the next step is manual error correction which is carried

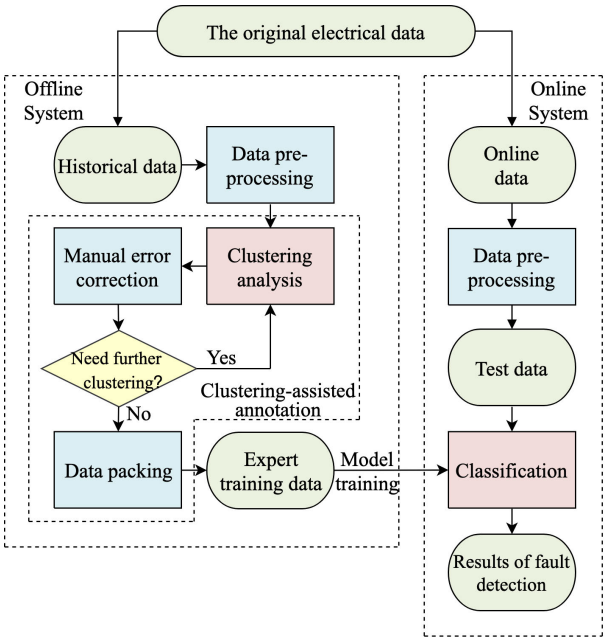


FIGURE 1. Flow chart of fault detection for spacecraft electronic load systems.

out to check whether the labels are correct. If there are subsets of data that need further clustering (e.g., two categories of signal are misclassified as one category), the subsets will be reprocessed by the clustering analysis and manual error correction in turn. Until all the signals are correctly labeled, the data are packed into expert training dataset, which is used to train the classification model. In the online system, the online data is pre-processed to be test data. Then, the test data is classified by the well-trained classification model which can output the results of fault detection.

In the process shown in Fig. 1, the performances of the clustering analysis algorithm and the classification algorithm are important factors that affect the quality and efficiency of data processing. For the high-dimensional signal data, the clustering analysis and classification algorithms should have excellent feature extraction ability to refine low-dimensional information from high-dimensional data. Therefore, we design a novel CNN module, the MRD module, to extract different scales of effective features from signal data effectively. The detail of the MRD module is introduced in Section III-A. Then, the MRD-CNN model is proposed for the signal classification task, which is described in Section III-B. Moreover, the MRD-CluVAE is presented for the signal clustering analysis task, which is introduced in Section III-C.

A. MRD MODULE

In the electrical signal data, complex features of multiple scales are included in the long time-series. To enable the network to capture features in different scales, one solution is to utilize convolutional filters with different sizes in the network architecture. However, in the naive sequential CNNs models, there are only filters with the same size in the same depth of networks. When the convolutional layers with various sizes of filters are stacked into a single sequence, the scale of the receptive fields in the same depth is constant [22].

For the extraction of multi-scale features, another solution is to increase the depth of networks. Unfortunately, when a CNN goes deep, the vanishing gradient problem will arise, which will lead to worse performance and poorer convergence [14]. To combat the vanishing gradient and enable the network to extract multi-scale features from long-time signals, specific network architecture or module need to be designed.

In this paper, we propose the MRD module, which is a kind of 1D-CNNs module for the time-series signal processing tasks. Having utilized and improved a variety of advanced techniques, the proposed module has strong feature extraction performance and robustness on the one-dimensional time-series electrical signals processing task. The architecture of the MRD module is shown in Fig. 2.

Inspired by the Inception module proposed by Szegedy *et al.* [15], [23], we design two branches with convolutional filters of different sizes to extract features with different scales. Specifically, in the large scale features extraction branch (LB, the red branch in Fig. 2) the sizes of the convolutional filters are 5, while in the small scale features extraction branch (SB, the blue branch in Fig. 2) the sizes of the convolutional filters are 3.

Moreover, to mitigate the vanishing gradient problem, a shortcut branch (the yellow branch in Fig. 2) is applied in the MRD module, which is inspired by the idea of ResNet in [14]. To restrict the channel numbers of the output feature maps, a convolutional layer (filter sizes equal to 1) without activation is utilized in the shortcut.

Proposed by Yu and Koltun, the dilated convolutions support expansion of the receptive field without loss of resolution or coverage [24]. In the presented module, one-dimensional dilated convolutions are applied in both LB and SB, which are indicated as the ''DilConv1D'' in Fig. 2. Specifically, the dilation rate of the second convolutional layers in the SB and the first convolutional layers in the LB is 2, while the dilation rate of the second convolutional layers in the LB is 5. The expansion of the receptive field can be illustrated in Fig. 3. By utilizing the dilated convolutions, the receptive field of LB (Fig. 3(a)) is extended to 25, while the receptive field of 2-layer CNN without dilated convolution (Fig. 3(c)) is 9. Similarly, the receptive field of SB increases from 5 (Fig. 3(d)) to 7 (Fig. 3(b)).

Meanwhile, the structures with a pair of parallel convolutional layers are widely employed in LB and SB, such as the convolutional layers connecting A1 and A2-1/A2-2 in Fig. 2. This aggregated structure is derived from the ResNeXt block and equivalent to the ResNeXt blocks with the cardinality equal to 2 [25]. The simple aggregated structure makes the network more adaptable and improve classification accuracy, while the complexity of the model is not increased.

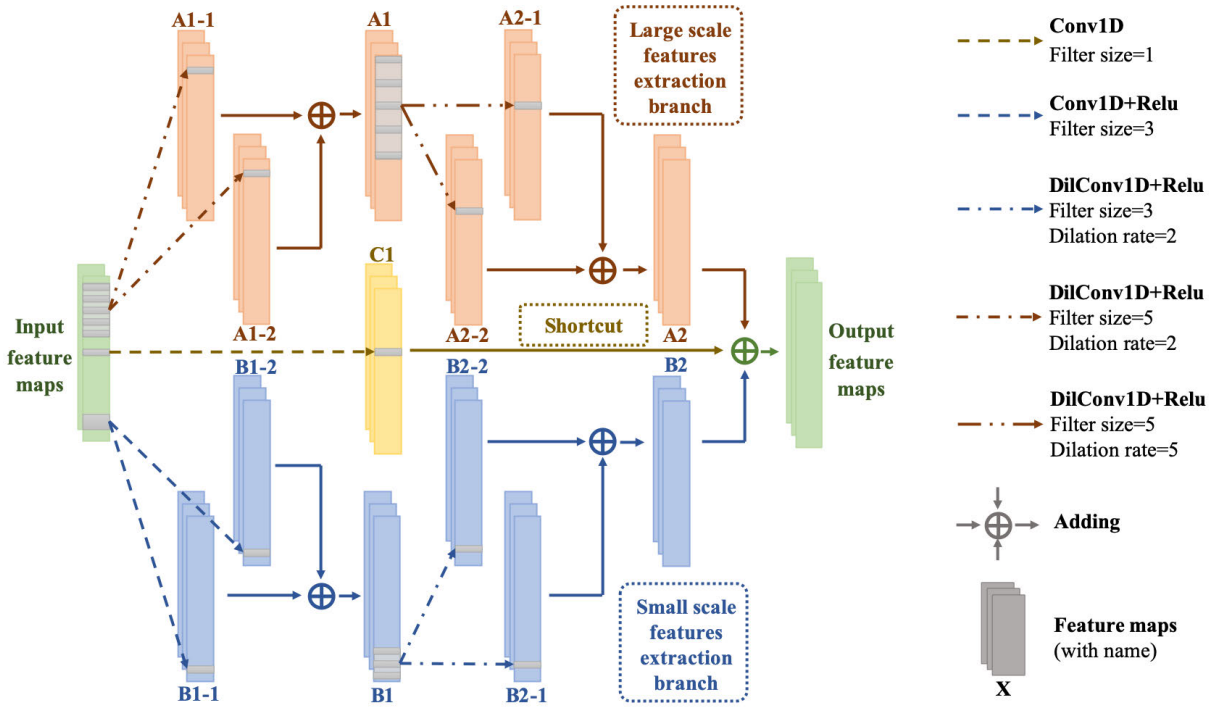
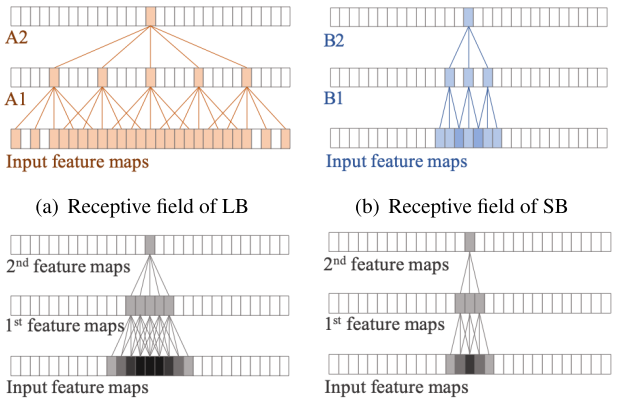


FIGURE 2. Architecture of the proposed MRD module.



(a) Receptive field of LB (b) Receptive field of SB

(c) Receptive field of 2-layer CNN wihout dilated convolution (filter size = 5) (d) Receptive field of 2-layer CNN wihout dilated convolution (filter size = 3)

FIGURE 3. The comparison of receptive fields of LB, SB and 2-layer CNNs without dilated convolution.

Different from most of the inception networks, adding operation is used for data fusion of different branches in the proposed module, replacing the concatenation layers. This design guarantees the multi-channel transmission of the information flow while restricting the number of network parameters.

B. MRD-CNN FOR SIGNAL CLASSIFICATION

By stacking the MRD modules, we construct the MRD-CNN for signal classification. The proposed network is mainly composed of two parts, feature extraction and classification. The feature extraction part is responsible for automatically extracting the effective features from the electrical signal, while the classification part is responsible for accurate signal classification using the extracted features [17]. The input data of the network is the pre-processed electrical signal, and the output is the predicted class of the input signal. The architecture of the proposed MRD-CNN is demonstrated in Fig. 4.

The feature extraction part is comprised of the MRD modules and down-sampling layers (max pooling layers). The MRD modules can extract features in different scales and avoid the gradient vanishing problem, while the max pooling layers can restrict the number of weights and prevent the overfitting problem. The flexible architecture of the feature extraction part can make the network more adaptable to specific classification problems. When the signal is complex and lengthy, the number of ''MRD module + max pooling layer'' units should be larger. When the data is simple, the unit number can be reduced to acquire a higher running speed.

After feature extraction, feature maps are output by the last down-sampling layer. Then, the feature maps should be vectored [26]. In the proposed model, a structure called global average pooling layer is utilized for the connection between the feature maps and the feature vector, rather than the traditional fully connected layer. The output of global average pooling layer can be described as

$$p_j^l = \frac{\sum_{i=1}^{L} x_i^{l-1,j}}{L}, \tag{5}$$

where l is the layer index, L is the length of the feature map, p_j^l is the j^{th} element of the output vector of the global average

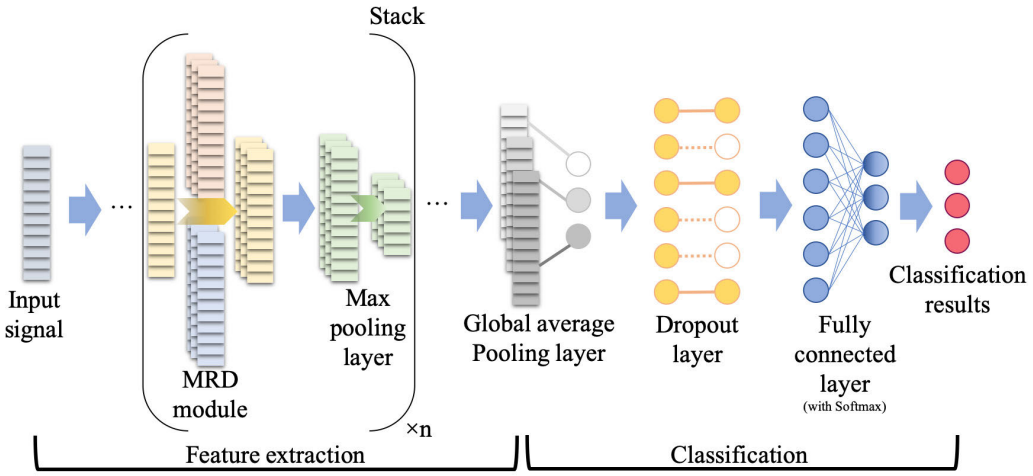


FIGURE 4. Architecture of the proposed MRD-CNN for signal classification.

pooling layer, and $x_i^{l-1,j}$ is the output of the $(l-1)^{th}$ layer for feature map j and sequence index i.

The advantage of global average pooling over the fully connected layers is that it is more native to the convolution structure by enforcing the correspondences between feature maps and categories [26]. Another advantage is that the global average pooling layer does not have any parameters, which means fewer computations and less likely overfit.

To prevent overfitting, we also employed the dropout layer before the final fully connected layer in the model [27], [28]. At the end of the classification part, there is a fully connected layer to output the multi-class classification result. Then, the output of the fully connected layer is followed by a softmax function, and the loss function is defined as the categorical cross-entropy loss.

C. MRD-CLUVAE FOR SIGNAL CLUSTERING ANALYSIS

For the clustering analysis of signal, inspired by Su's work in [29] and [30], the MRD-CluVAE is presented. First, the joint probability distribution is directly used to define the loss function of VAE and clustering VAE (CluVAE). Then, the architecture of VAE is modified, so that the VAE-based model can output the clustering results directly. Moreover, the MRD modules are applied to the encoder network and decoder network of the VAE model.

1) JOINT PROBABILITY DISTRIBUTION

The principle of VAE focuses on the posterior distribution $p(z|x)$ which is difficult to compute in some situation. Actually, it is more convenient to directly use $q(x, z)$ to approximate $p(x, z)$ and define the loss function of VAE as $\mathrm{KL}(q(x, z)||p(x, z))$.

$$\begin{aligned} L_{\mathrm{VAE}} &= \mathrm{KL}(q(x,z)\|p(x,z)) \\ &= \mathrm{E}_{x\sim q(x)}\mathrm{E}_{z\sim q(z|x)}[\log q(x)] \\ &\quad + \mathrm{E}_{x\sim q(x)}[\mathrm{KL}(q(z|\mathrm{x})||p(z)) \\ &\quad - \mathrm{E}_{z\sim q(z|x)}[\log p(x|z)]] \end{aligned} \tag{6}$$

In (6), $q(x)$ is the prior distribution over x. It is definite even if we cannot obtain its concrete formula. So the first term $\mathrm{E}_{x\sim q(x)}\mathrm{E}_{z\sim q(z|x)}[\log q(x)]$ is a constant. While the second term $\mathrm{E}_{x\sim q(x)}[\mathrm{KL}(q(z|x)||p(z)) - \mathrm{E}_{z\sim q(z|x)}[\log p(x|z)]]$ is an expected traditional VAE loss function as is introduced in (3). The deduction of (6) is given in 16 in Appendix A.

2) MRD-CLUVAE

In VAE, the latent variable is denoted as z, which can be either continuous or discrete. For any clustering issues, each sample belongs to an explicit category y. CluVAE is proposed to solve unsupervised classification problems. In CluVAE, the latent variable can be set as (z, y), where z is a continuous encoding vector and y is a discrete vector. Inspired by (6), the loss function for CluVAE is defined as

$$\begin{aligned} L_{\mathrm{CluVAE}} &= \mathrm{KL}(p(x,y,z)\|q(x,y,z)) \\ &= E_{x\sim p(x)}\mathrm{E}_{z\sim p(z|x)}[-\log q(x|z) \\ &\quad + \mathrm{KL}(p(y|z)||q(y)) + \sum_y p(y|z)\log\frac{p(z|x)}{q(z|y)}]. \end{aligned} \tag{7}$$

The deduction (7) is given in 17 in Appendix A.

In (7), the first term $-\log q(x|z)$ refers to the reconstruction error. (In practice, it can be replaced by the mean square error of the encoder's input x and decoder's output x'.) The second term helps to achieve a balanced distribution which is optional. The third term $\sum_y p(y|z)\log\frac{p(z|x)}{q(z|y)}$ plays a key role in clustering. In the third term,

$$p(z|x) = \frac{1}{\prod_{i=1}^{d}\sqrt{2\pi\sigma_i^2(x)}} e^{-\frac{1}{2}\left\|\frac{z-\mu(x)}{\sigma(x)}\right\|^2}, \tag{8}$$

and

$$q(z|y) = \frac{1}{(2\pi)^{d/2}} e^{\left\{-\frac{1}{2}\|z-\mu_y\|^2\right\}}. \tag{9}$$

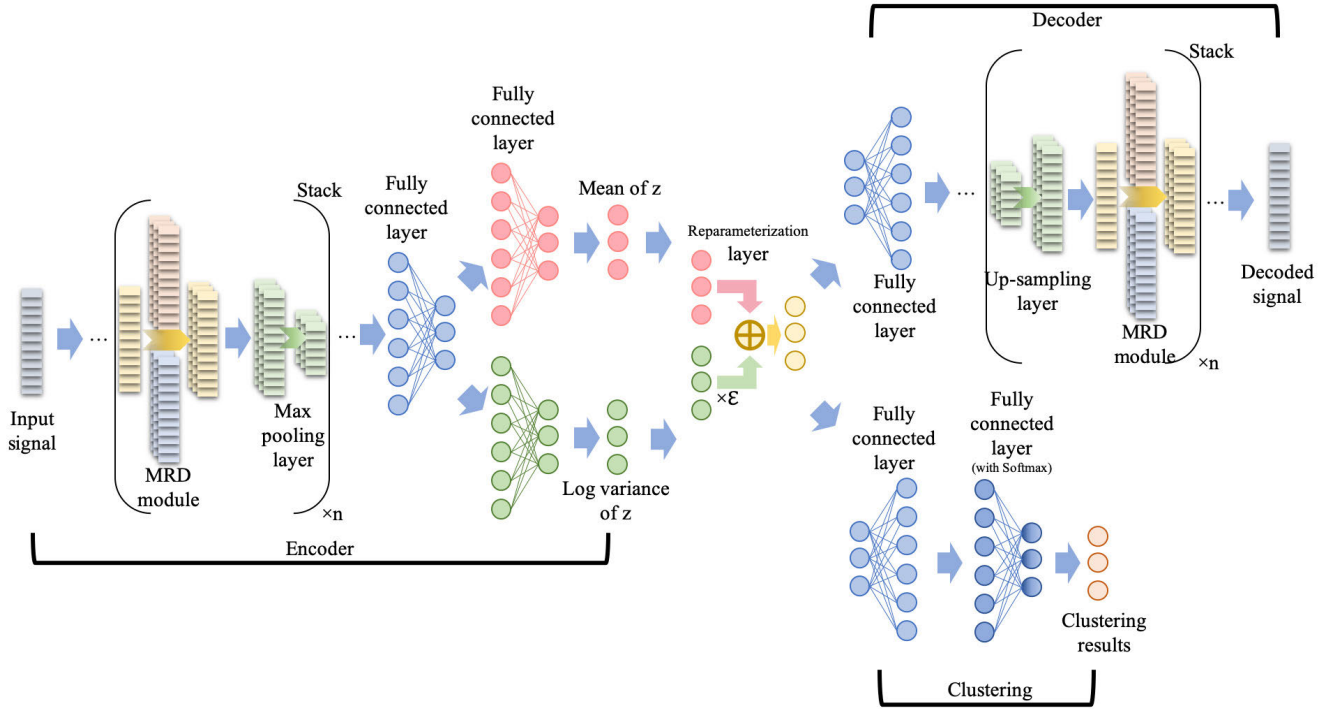


FIGURE 5. Architecture of the proposed MRD-CluVAE for signal clustering.

So

$$\log \frac{p(z|x)}{q(z|y)} = -\frac{1}{2}\sum_{i=1}^{d}\log\sigma_i^2(x) - \frac{1}{2}\left\|\frac{z-\mu(x)}{\sigma(x)}\right\|^2 + \frac{1}{2}\left\|z-\mu_y\right\|^2. \quad (10)$$

In (10), $\frac{1}{2}\left\|\frac{z-\mu(x)}{\sigma(x)}\right\|^2$ can be ignored because of reparameterization. During reparameterizing, ε is sampled from (0, 1) and make $z = \mu(x) + \varepsilon \times \sigma(x)$. So the second term of (10) equals to $-\|\varepsilon\|^2/2$, actually.

The architecture of the proposed MRD-CluVAE for signal clustering is demonstrated in Fig. 5. First, the input signal x is fed into the encoder which outputs the latent vector z according to $p(z|x)$. Similar to the traditional VAE structure, a decoder networks with a symmetrical architecture inputs the reparameteried z and outputs the reconstruction of the input signal x. In particular, the MRD modules (proposed in Section III-A) are applied to the network of encoder and decoder and replace the fully connected layers in traditional VAE.

Then, to get the clustering results, z is used as an input to a clustering network which is composed of fully connected layers in our model. After the last fully connected layer, a softmax activation is followed. According to $p(y|z)$, the clustering network outputs the category y which x belongs to. $p(z|x)$ is assumed to be a normal distribution with mean $\mu(x)$ and variable $\sigma^2(x)$. This distribution is determined by the neural network. $q(z|y)$ is assumed to be another normal distribution with mean $\mu(y)$ and $\sigma^2(y)$.

IV. EXPERIMENTS

In this paper, a widely used DL framework Keras [31] with TensorFlow [32] as backend has been used for developing and training the proposed models. The experiments have been performed on a personal computer with Ubuntu 16.04, an Intel Core i7-7700K (4.20 GHz) CPU, an NVIDIA GeForce GTX 1080Ti GPU and 15.6 GB of RAM.

A. MRD-CNN-BASED SIGNAL CLASSIFICATION

1) DATASET

The experimental data comes from the typical electrical characteristics data of spacecraft electronic load systems. After pre-processing, a total of 22800 signal samples with 19 different types are acquired, and each signal sample contains 1000 data points. In this paper, the dataset is abbreviated as SELS dataset. Specifically, the waveforms of the signals are demonstrated in Fig. 6.

For classification, the dataset with 22800 signal samples is divided into training set and test set. 11400 signal samples (50% of the total) are used for the training phase of the model, and 11400 signal samples are used for testing the performance of the classifier model. To fine-tune the parameters of the classifier model, 1400 samples from the training set are picked as the validation set, and the remaining 10000 samples are employed to train the MRD-CNN model.

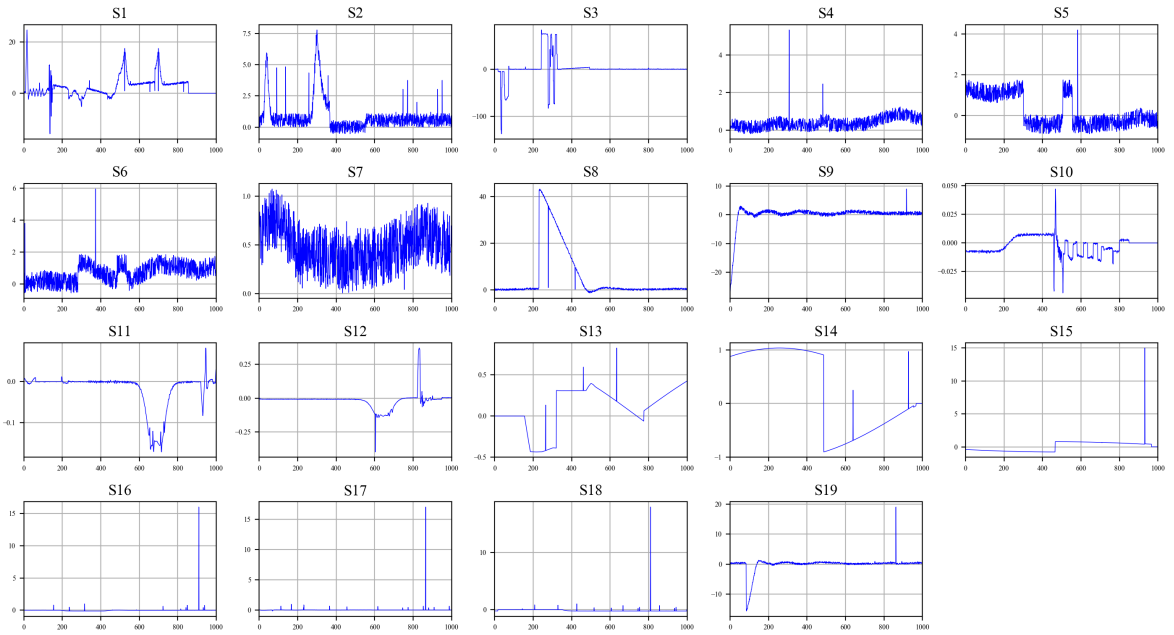


FIGURE 6. Example signals from each class in the SELS dataset.

2) DETAIL FOR MODEL AND TRAINING

As described in Section III-B, the MRD-CNN model consisting of the MRD modules, max pooling layers, a global average pooling layer, a dropout layer, and a fully connected layer. For the specific task, several key parameters about the network architecture, including the number of layers and the number of output channels for each MRD module, are tested. We define an "MRD module + max pooling layer" block as an essential unit of the network, and the number of output channels for a unit is defined as C. Then, five types of network architectures are tested on the validation dataset, and the relation between validation accuracy of different architecture and the value of C is obtained and shown in Fig. 7. Considering factors such as validation accuracy and network size, the 6-unit architecture "16-16-32-32-64-64" is chosen for the classification of SELS dataset.

Specifically, the detail of the model is shown in Table. 1. For the specific task, 6 MRD modules with max pooling layers are stacked into the model. As the network goes deeper, the number of channels is increasing. Furthermore, the output of the fully connected layer is followed by the softmax function with the categorical cross-entropy loss function. To sum up, there are 15 layers/modules in the model, and the total number of parameters is 310,467.

After fine-tuning, the hyper-parameters for the model and training process are selected and given in Table. 2. According to [33], the dropout probability is set to 0.5 to maximize the number of randomly generated subnetworks. The network is trained by the Adam optimization method [34], which is a robust and effective training optimizer for neural networks. For the Adam optimization method, after validation experiments, the learning rate α is set to be 0.001, and the values of β_1 and β_2 are set to be 0.9 and 0.999, respectively. Moreover, the size of minibatch is 128, and the number of training epoch is set to be 50, which ensures the convergence of the loss function.

TABLE 1. Details of the MRD-CNN model for classification of SELS dataset.

Layer	Output Size	Number of Output Channels	Number of Parameters
Input layer	1000	1	0
The MRD module	1000	16	4,396
Max pooling layer	500	16	0
The MRD module	500	16	8,592
Max pooling layer	250	16	0
The MRD module	250	32	25,376
Max pooling layer	125	32	0
The MRD module	125	32	34,080
Max pooling layer	62	32	0
The MRD module	62	64	100,928
Max pooling layer	31	64	0
The MRD module	31	64	135,744
Max pooling layer	15	64	0
Global average pooling layer	64	-	0
Dropout layer	64	-	0
Fully connected layer	19	-	1235

TABLE 2. Hyper-parameters of the MRD-CNN model for classification of SELS dataset.

Parameter	Value
Dropout probability	0.5
Learning rate α	0.001
β_1 of Adam optimizer	0.9
β_2 of Adam optimizer	0.999
Adam minibatch size	128
Max epochs	50

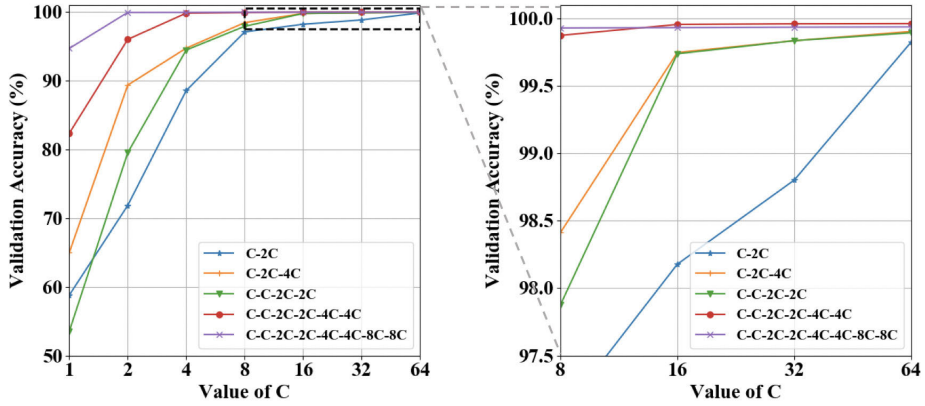


FIGURE 7. Results of the optimization experiments for the MRD-CNN's architecture.

3) CLASSIFICATION RESULTS AND DISCUSSION

The performance of the proposed MRD-CNN model is measured using four standard metrics: the average precision (AP), the average recall (AR), the accuracy (ACC) and the F1 measure. They are calculated as

$$AP = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FP_i}, \tag{11}$$

$$AR = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_1}{TP_i + TN_i}, \tag{12}$$

$$ACC = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i + TN_i}{TP_i + FP_i + FN_i}, \tag{13}$$

$$F1\text{measure} = 2 \times \frac{AP \times AR}{AP + AR}, \tag{14}$$

where TP_i represents the true positive and indicates the number of signals from class i that are correctly classified, FP_i represents the false positive and indicates the number of signals that are not from class i but are misclassified as class i, TN_i represents the true negative and indicates the number of signals that are incorrectly classified to other class rather than class i, FN_i represents the false negative and indicates the signals from class i but are misclassified, and N is the number of classes. For the SELS dataset, the value of N is 19.

To cross-validate, 20 times of experiments are carried out. For each experiment, fifty percent of the signal data is randomly selected as the training set, and the rest is used as the test set. After experiments, the mean values and standard deviations of AP, AR, ACC and F1 measure are $99.90 \pm 0.04\%$, $99.90 \pm 0.03\%$, $99.90 \pm 0.04\%$ and $99.90 \pm 0.03\%$, respectively. From the results of the 20 tests, one of the test results that can represent the average performance of the proposed model (the evaluation metrics are 99.90%, 99.89%, 99.89% and 99.90%, respectively) is selected to analyze the performance of the proposed model.

To specifically appraise the classification results, a confusion matrix is utilized to evaluate the performance of the proposed model, as is shown in Fig. 8. Here, 17 categories of signals are correctly classified. Only a few samples of category S16 (0.68% of the total number of samples from class S16) and S17 (0.35% of the total number of samples from class S17) are misclassified as other categories due to the similarity of the waveforms of the signals.

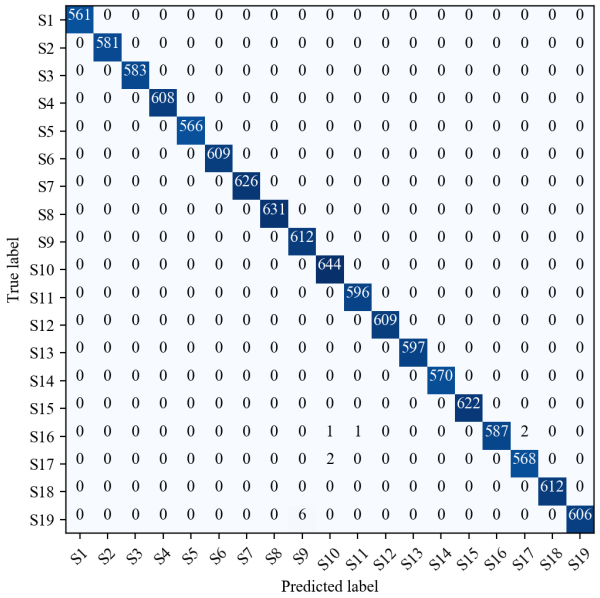


FIGURE 8. The confusion matrix of the classification with 19 categories using the MRD-CNN.

For comparison, K-Nearest Neighbor (KNN), SVM and RF are selected as the comparative algorithms. In the SVM algorithm, the Gaussian radial basis function (RBF) is employed as the kernel function. In the RF algorithm, the number of decision trees is set to be 100. Meanwhile, the previously proposed DBN-RF [2] and DNN-WSVM [35] methods are also utilized for comparison purposes. Moreover, a sequential CNN classification model and two advanced DL models for time series classification, the ResNet (a one-dimensional residual network) [36] and the LSTM-FCN [37],

TABLE 3. The comparison of the classification performances using different methods.

Method	AP	AR	ACC	F1 measure
KNN	$79.51 \pm 1.08\%$	$84.82 \pm 0.23\%$	$84.77 \pm 0.31\%$	$81.24 \pm 0.23\%$
SVM	$91.14 \pm 0.64\%$	$89.30 \pm 1.16\%$	$89.27 \pm 1.20\%$	$89.13 \pm 1.19\%$
RF	$94.70 \pm 0.11\%$	$94.69 \pm 0.11\%$	$94.68 \pm 0.10\%$	$94.68 \pm 0.10\%$
DNN-WSVM [36]	$98.60 \pm 0.15\%$	$98.61 \pm 0.16\%$	$98.59 \pm 0.14\%$	$98.57 \pm 0.15\%$
DBN-RF [2]	$98.95 \pm 0.09\%$	$98.95 \pm 0.09\%$	$98.94 \pm 0.10\%$	$98.95 \pm 0.09\%$
Sequential CNN	$98.88 \pm 0.25\%$	$98.71 \pm 0.33\%$	$98.69 \pm 0.35\%$	$98.67 \pm 0.37\%$
ResNet [37]	$98.66 \pm 0.72\%$	$98.31 \pm 1.09\%$	$98.30 \pm 1.11\%$	$98.24 \pm 1.19\%$
LSTM-FCN [38]	$98.99 \pm 0.11\%$	$98.97 \pm 0.12\%$	$98.98 \pm 0.11\%$	$98.98 \pm 0.11\%$
MRD-CNN	$\mathbf{99.90 \pm 0.04\%}$	$\mathbf{99.90 \pm 0.03\%}$	$\mathbf{99.90 \pm 0.04\%}$	$\mathbf{99.90 \pm 0.03\%}$

are included for comparison. Specifically, the CNN model has 12 one-dimensional convolutional layers, and the total number of parameters is 1, 247, 955. In the ResNet, the numbers of filters for each block are set to be 64, 128 and 128 respectively according to [36], while the total number of parameters is 2, 863, 831. For the LSTM-FCN, the number of cells is set to be 64, and the total number of parameters is 542, 035. All the baseline models are well-trained to show their best performances. Similarly, the same dataset is randomly divided into a training set and a test set. After 20 tests for each classifier, the AP, AR, ACC and F1 measure are recorded as the evaluation metrics.

The comparison of the classification performances using different classifiers is demonstrated in Table. 3. As shown in the table, the performance of the KNN algorithm is poor. Other traditional shallow learning methods, such as SVM and RF, have better performance than KNN. The metrics of methods combining feature extraction and a classification algorithm, such as the DNN-WSVM and DBN-RF, are over 98%, which are higher than the traditional classification algorithms. Meanwhile, the sequential CNN shows a similar performance with the combining algorithms but has poorer stability than those methods (the standard deviations of the four metrics are over 0.2%). Although the ResNet got competitive performances on several time series classification tasks in [36], for our dataset, this method shows instability and over-fitting problem in the process of training, and its classification performance is lower than sequential CNN. The results of LSTM-FCN are more exhilarating than other baseline approaches, but there is room for improvement. Compared with other algorithms, the proposed MRD-CNN method achieves higher performance and stability for each metric. Specifically, the average values of the evaluation metrics are 99.90% and the standard deviations are less than 0.05%.

As the comparison shown, the performance of the traditional classification algorithms is limited due to the difficulty in processing the high-dimensional signal data. Then, combined with feature extraction algorithms, the performance of these traditional algorithms are improved to some extent. By automatic feature extraction, the sequential CNN can achieve a similar performance. The two deep learning models for time series classification have more complex architectures, but have worse performances than the MRD-CNN. Moreover, the training processes of ResNet and LSTM-FCN are more time-consuming than our model. The reason why the proposed method is superior to other algorithms is that it can extract multi-scale features automatically and learn deeply without the influence of the gradient vanishing problem. On the one hand, by the introduction of multi-branch modules with dilated convolutions, the multi-scale features are extracted efficiently. On the other hand, the shortcut branch enables the model to suppress gradient vanishing problem when the network is deep.

B. MRD-CLUVAE-BASED SIGNAL CLUSTERING ANALYSIS

1) DATASET

In the phase of clustering analysis experiments, two datasets are utilized to test the performance of the clustering analysis algorithm. The first dataset is the SELS dataset, which has been introduced in Section IV-A.1. The second dataset is the randomized battery usage dataset [38] from the Prognostics Center of Excellence (PCoE) of National Aeronautics and Space Administration (NASA). Thousands of voltage records of charging and discharging cycles of batteries are included in the dataset. After manual segmentation and labeling, a total of 1591 samples were selected as the data for algorithm verification. The length of the signal samples is 500, and the number of categories is 7. Besides, the number of the seven categories is imbalanced. The category with most data contains 370 samples, while the category with least data contains 70 samples. Specifically, the waveforms of the signals are elucidated in Fig. 9.

For both of the datasets, the signal samples are divided into training set and test set. Specifically, the training set has 80% of data and the test set has 20%. Different from the classification task, the labels of the training sets are discarded, which means the input data of the MRD-CluVAE clustering analysis model is unlabeled raw data in the training phase. When the clustering analysis model is well-trained, the test data with labels is employed to evaluate the performance of the model.

2) DETAIL FOR MODEL AND TRAINING

As described in Section III-C, the MRD modules are adopted as the encoder and decoder in the CluVAE network. Specifically, for the encoder network, two MRD modules followed

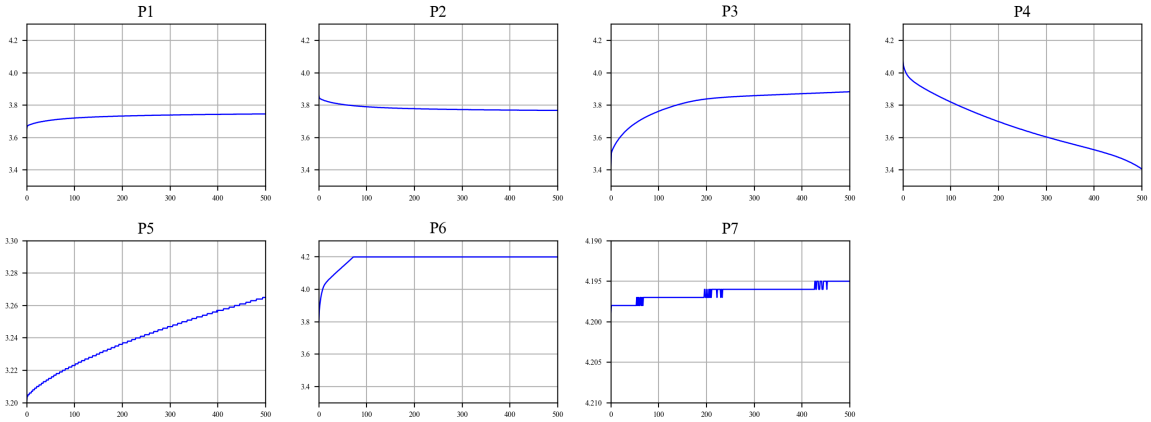


FIGURE 9. Example signals from each class in the PCoE dataset.

TABLE 4. Hyper-parameters of the MRD-CluVAE model for clustering analysis.

Parameters	SELS dataset	PCoE dataset
Dimension of the latent vector	16	50
Epoch number of the first stage	300	500
Learning rate of the first stage	0.001	0.001
Epoch number of the second stage	300	500
Learning rate of the second stage	0.0001	0.0001
Epoch number of the third stage	100	-
Learning rate of the third stage (SGD)	10^{-5}	-
β_1 of Adam optimizer	0.9	0.9
β_2 of Adam optimizer	0.999	0.999
Adam minibatch size	128	128

by max pooling layers are used, and the number of output channels for the two modules are 32 and 16, respectively. For the decoder network, a symmetrical structure with the encoder is utilized: two MRD modules after up-sampling layers are constructed into the decoder, and the numbers of output channels are 16 and 32, respectively.

The hyper-parameters of the MRD-CluVAE are given in Table. 4. The dimension of the latent vector in the model should be set according to the specific dataset. For the SELS dataset, this hyper-parameter is set to be 16, while that of the PCoE dataset is set to be 50. To make the model well-trained, multi-stage training is carried out for the MRD-CluVAE. For the SELS dataset, a three-stage training is utilized. the learning rate of each stage is set to be 0.001, 0.0001 and 10^{-5} respectively, and the epoch numbers are 300, 300 and 100 respectively. Especially, instead of Adam optimizer, an SGD optimizer is applied to the third training stage. For the PCoE dataset, a two-stage training is employed, where the learning rate is 0.001 and 0.0001 respectively and the epoch number is 500 for both stages. The other hyper-parameters of the Adam optimizer are the same as the classification phase.

3) CLUSTERING ANALYSIS RESULTS AND DISCUSSION

To measure the performance of the clustering analysis, a standard unsupervised evaluation metric, the unsupervised clustering accuracy [39], is used as the evaluation metric. The clustering accuracy is described as

$$ClusteringACC = \max_{m} \frac{\sum_{i=1}^{n} \mathbf{1}\{l_i = m(c_i)\}}{n}, \quad (15)$$

where l_i is the ground-truth label, c_i is the cluster assignment produced by the clustering algorithm, and m ranges over all possible one-to-one mappings between clusters and labels.

For comparison, we test the performance of some typical clustering analysis algorithms on our datasets. Two classic clustering algorithms, K-means and DBSCAN, are adopted as contrasts. The methods combine AE/VAE and K-means are also included. Meanwhile, two advanced deep clustering approaches are introduced for comparison, which are the DEC [39] and SpectralNet [40]. In the DEC, the epoch numbers for pretraining and clustering are set to be 500 and 1000 respectively for the SERS dataset, while they are set to be 120 and 300 respectively for the PCoE dataset. For the SpectralNet, we pretrain autoencoders for each dataset, and the number of training epochs for the Siamese network and spectral network is 400 for the SERS dataset, and 200 for the PCoE dataset. Also, the initial learning rate of the spectral network is set to be 0.0001 to prevent the calculation errors of Cholesky decomposition. In addition, we also change the encoder/decoder networks in the MRD-CluVAE from the MRD modules to normal DNN and CNN, and these methods are described as DNN-CluVAE and CNN-CluVAE respectively.

For the MRD-CluVAE and the baseline algorithms, we perform 20 times of experiments and pick the result with the best metric value. Especially, if the algorithms need the number of clusters as a hyper-parameter (such as K-means and CluVAE), the number of the ground-truth categories will be given. For all the algorithms, we fine-tune the hyper-parameters carefully to guarantee the best performance.

The comparison of clustering performances on the two datasets are demonstrated in Table. 5. As shown in the table, the traditional clustering algorithms, such as K-means and DBSCAN, do not achieve a good clustering performance,

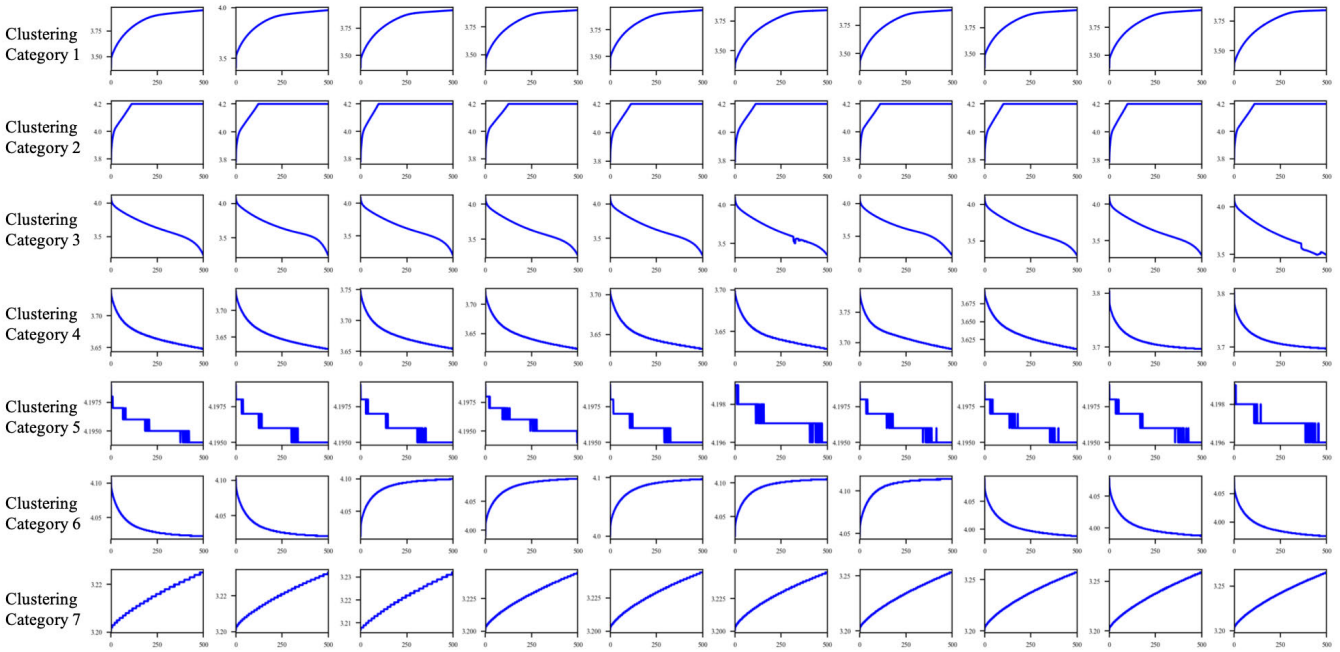


FIGURE 10. Signals with 10 top clustering scores from each clustering category on PCoE dataset.

TABLE 5. The comparison of the clustering performances using different methods.

Method	Clustering ACC on SELS dataset	Clustering ACC on PCoE dataset
K-means	25.35%	54.99%
DBSCAN	18.73%	57.37%
AE+K-means	27.15%	58.03%
VAE+K-means	30.38%	64.97%
DEC [40]	34.90%	78.94%
SpectralNet [41]	49.70%	70.31%
DNN-CluVAE	61.63%	74.68%
CNN-CluVAE	60.52%	67.52%
MRD-CluVAE	**71.07%**	**84.14%**

especially when the number of categories is large. When combining with deep features extraction methods, the K-means algorithm shows a little progress. The classic deep clustering method, DEC, results in a competitive performance on the PCoE dataset, but cannot cluster well on the SERS dataset. By comparison, SpectralNet has a better performance on the complex SERS dataset. Compared with the clustering performance of other algorithms, the results of CluVAE methods are significantly improved, especially for the SERS dataset. Among them, the clustering performance of the DNN-CluVAE is slightly better than that of the CNN-CluVAE, and the MRD-CluVAE has the highest clustering performance. Specifically, the MRD-CluVAE performs a clustering ACC of 84.14% on the PCoE dataset a clustering ACC of 71.07% on the SELS dataset.

For the high-dimensional signal data, the traditional clustering algorithms cannot perform well due to the lack of feature extraction. However, when combined with deep features extraction algorithms, the progress is still limited since the gap between the feature extraction and clustering algorithms. To solve this problem, many researchers are seeking for an end-to-end clustering solution based on deep learning, such as DEC and SpectralNet. However, in these methods, the autoencoder networks are pretrained before they are utilized for the follow-up clustering works, which means the training processes of autoencoders and clustering networks are separate. Different from those methods, the CluVAE networks combining autoencoder with a clustering branch can output clustering results directly. In the CluVAE, the training phases of the autoencoder and clustering branch are simultaneous. Therefore, the features extracted by the encoder network are related to the clustering output, and that is the reason why the CluVAEs have superior clustering performance. Meanwhile, the excellent feature extraction capability of the MRD modules enables CluVAE to extract features from the raw signals and reconstruct the signals accurately. Thus, the MRD-CluVAE achieves a better performance than other CluVAE methods.

In Fig. 10, the signals with 10 top clustering scores from each clustering category on PCoE dataset are displayed. Each row corresponds to a clustering category, and the signals are sorted from left to right based on their output scores of CluVAE. As shown in Fig. 10, most clustering categories correspond to natural categories very well. For example, the clustering category 2,3,5 are homologous with P6, P4 and P7 in Fig. 9, respectively. Exceptionally, the samples in the cluster category 6 are cluttered since some of the signals are misclassified. The reason is that the model extracts the amplitude of signals as a more important feature, instead of the gradient.

V. CONCLUSION

For the fault detection of spacecraft electronic load systems, the recognition of electrical signal plays a key role in the systems. Therefore, a classification algorithm with great performance and a clustering analysis algorithm with high clustering accuracy are urgently needed for the processing of signal. For the construction of DL models, the structure for feature extraction is of great importance.

In this paper, a one-dimensional convolutional module, the MRD module, is introduced. By using advanced techniques such as multi-scale branches and dilated convolutions, the module can extract multi-scale features from high dimensional signal data. At the same time, the residual shortcut structure in the model suppresses the problem of gradient vanishing. Then, we can make the network deeper. Since the flexible architecture of neural networks, the MRD module can be applied to both classification models and clustering models.

Based on the MRD module, we construct a classification model, the MRD-CNN, for the classification task of electrical signal. In the proposed model, advanced methods are applied to obtain a better classification performance and higher training efficiency. The experimental results of the classification demonstrate that the proposed model has higher classification accuracy and robustness when dealing with the signal classification problem.

Moreover, to optimize the process of data annotation in the offline system, we suggest a clustering analysis model by modifying the architecture of VAE and combining it with the MRD module. In the experiment phase, the proposed MRD-CluVAE model achieves a better clustering performance than the baseline algorithms, and the visualization of the clustering categories indicates that the proposed model can categorize most of the signals into the correct categories.

Future work will include the extension of the current MRD-CNN model and MRD-CluVAE model for more complex one-dimensional data processing tasks, such as hyperspectral data and ECG data. In addition, since the VAE-based clustering models are effective for the clustering analysis problems, more research about the MRD-CluVAE and similar models will be carried out in future.

APPENDIX

A. THE DEDUCTION OF LOSS FUNCTIONS OF VAE AND CLUVAE

$$
\begin{aligned}
L_{\mathrm{VAE}} &= \mathrm{KL}(q(x,z)\|p(x,z)) = \iint q(x,z)\log\frac{q(x,z)}{p(x,z)}dxdz \\
&= \iint q(z|x)q(x)\log\frac{q(z|x)q(x)}{p(x|z)p(z)}dxdz \\
&= \iint q(z|x)q(x)(\log q(x)+\log\frac{q(z|x)}{p(x|z)p(z)})dxdz \\
&= \iint q(z|x)q(x)\log q(x)dxdz \\
&\quad + \iint q(z|x)q(x)\log\frac{q(z|x)}{p(x|z)p(z)}dxdz \\
&= \mathrm{E}_{x\sim q(x)}\mathrm{E}_{z\sim q(z|x)}[\log q(x)] \\
&\quad + \iint q(z|x)q(x)\log\frac{q(z|x)}{p(z)}dxdz \\
&\quad - \iint q(z|x)q(x)\log p(x|z))dxdz \\
&= \mathrm{E}_{x\sim q(x)}\mathrm{E}_{z\sim q(z|x)}[\log q(x)] \\
&\quad + \mathrm{E}_{x\sim q(x)}[\mathrm{KL}(q(z|x)||p(z)) \\
&\quad - \mathrm{E}_{z\sim q(z|x)}[\log p(x|z)]] \\
&= c_1 + \mathrm{E}_{x\sim q(x)}[\mathrm{KL}(q(z|x)||p(z)) \\
&\quad - \mathrm{E}_{z\sim q(z|x)}[\log p(x|z)]] \qquad (16)
\end{aligned}
$$

$$
\begin{aligned}
L_{\mathrm{CluVAE}} &= \mathrm{KL}(p(x,y,z)\|q(x,y,z)) \\
&= \sum_y \iint p(x,y,z)\log\frac{p(x,y,z)}{q(x,y,z)}dxdz \\
&= \sum_y \iint p(y|z)p(z|x)p(x)\log\frac{p(y|z)p(z|x)p(x)}{q(x|z)q(z|y)q(y)}dxdz \\
&= \sum_y \iint p(y|z)p(z|x)p(x)(\log p(x)-\log q(x|z) \\
&\quad + \log\frac{p(y|z)}{q(y)}+\log\frac{p(z|x)}{q(z|y)})dxdz \\
&= \mathrm{E}_{x\sim p(x)}\mathrm{E}_{z\sim p(z|x)}[\log p(x)-\log q(x|z) \\
&\quad + \mathrm{KL}(p(y|z)||q(y))+\sum_y p(y|z)\log\frac{p(z|x)}{q(z|y)}] \qquad (17)
\end{aligned}
$$

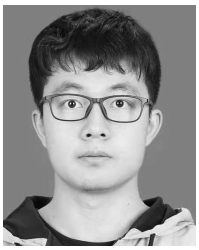
REFERENCES

[1] K. Xu and D. Pi, "A data-driven method of health monitoring for spacecraft," *Aircr. Eng. Aerosp. Technol.*, vol. 90, no. 2, pp. 435–451, 2018.
[2] K. Li, N. Yu, P. Li, S. Song, Y. Wu, Y. Li, and M. Liu, "Multi-label spacecraft electrical signal classification method based on DBN and random forest," *PLoS ONE*, vol. 12, no. 5, 2017, Art. no. e0176614.
[3] K. Li, Y. Liu, Q. Wang, Y. Wu, S. Song, Y. Sun, T. Liu, J. Wang, Y. Li, and S. Du, "A spacecraft electrical characteristics multi-label classification method based on off-line FCM clustering and on-line WPSVM," *PLoS ONE*, vol. 10, no. 11, 2015, Art. no. e0140395.
[4] Y. Liu, K. Li, Y. Huang, J. Wang, S. Song, and Y. Sun, "Spacecraft electrical characteristics identification study based on offline FCM clustering and online SVM classifier," in *Proc. Int. Conf. Multisensor Fusion Inf. Integr. Intell. Syst. (MFI)*, Sep. 2014, pp. 1–4.
[5] C. A. Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Syst. Appl.*, vol. 59, pp. 235–244, Oct. 2016.
[6] D.-T. Hoang and H.-J. Kang, "Convolutional neural network based bearing fault diagnosis," in *Proc. Int. Conf. Intell. Comput.* Springer, 2017, pp. 105–111.
[7] L. Jing, M. Zhao, P. Li, and X. Xu, "A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox," *Measurement*, vol. 111, pp. 1–10, Dec. 2017.
[8] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, Apr. 2017.
[9] N. Dilokthanakul, P. A. M. Mediano, M. Garnelo, M. C. H. Lee, H. Salimbeni, K. Arulkumaran, and M. Shanahan, "Deep unsupervised clustering with Gaussian mixture variational autoencoders," 2016, *arXiv:1611.02648*. [Online]. Available: http://arxiv.org/abs/1611.02648
[10] A. Ng, "Sparse autoencoder," *CS294A Lect. Notes*, vol. 72, pp. 1–19, 2011.
[11] Z. Jiang, "Variational deep embedding: A generative approach to clustering," *CoRR*, vol. abs/1611.05148, 2016. [Online]. Available: http://arxiv.org/abs/1611.05148
[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

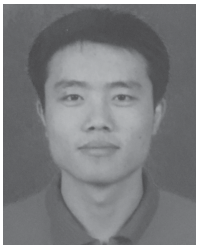
[13] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3128–3137.
[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
[16] L. Eren, "Bearing fault detection by one-dimensional convolutional neural networks," *Math. Problems Eng.*, vol. 2017, Jul. 2017, Art. no. 8617315.
[17] L. Dan, J. Zhang, Q. Zhang, and X. Wei, "Classification of ECG signals based on 1D convolution neural network," in *Proc. IEEE Int. Conf. e-Health Netw.*, Oct. 2017, pp. 1–6.
[18] D. P. Kingma and W. Max, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*. [Online]. Available: http://arxiv.org/abs/1312.6114
[19] M. I. Jordan and M. J. Wainwright, "Graphical models, exponential families, and variational inference," *Found. Trends Mach. Learn.*, vol. 1, nos. 1–2, pp. 1–305, 2007.
[20] C. K. Sønderby, T. Raiko, L. Maaløe, S. K. Sønderby, and O. Winther, "How to train deep variational autoencoders and probabilistic ladder networks," in *Proc. 33rd Int. Conf. Mach. Learn.*, 2016, pp. 1–10.
[21] E. Nalisnick, L. Hertel, and P. Smyth, "Approximate inference for deep latent Gaussian mixtures," in *Proc. NIPS Workshop Bayesian Deep Learn.*, vol. 2, 2016, pp. 1–4.
[22] X. Cao, "A practical theory for designing very deep convolutional neural networks," Tech. Rep., 2015.
[23] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-V4, Inception-ResNet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4278–4284.
[24] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*. [Online]. Available: http://arxiv.org/abs/1511.07122
[25] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1492–1500.
[26] M. Lin, Q. Chen, and S. Yan, "Network in network," 2013, *arXiv:1312.4400*. [Online]. Available: http://arxiv.org/abs/1312.4400
[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[28] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[29] J. Su, "Variational inference: A unified framework of generative models and some revelations," 2018, *arXiv:1807.05936*. [Online]. Available: http://arxiv.org/abs/1807.05936
[30] J. Su and G. Wu, "f-VAEs: Improve VAEs with conditional flows," 2018, *arXiv:1809.05861*. [Online]. Available: http://arxiv.org/abs/1809.05861
[31] F. Chollet. (2015). *Keras*. [Online]. Available: https://github.com/fchollet/keras
[32] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, and S. Ghemawat. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. [Online]. Available: http://tensorflow.org/
[33] P. Baldi and P. J. Sadowski, "Understanding dropout," in *Proc. Adv. Neural Inf. Process. Syst. 26*. Red Hook, NY, USA: Curran Associates, 2013, pp. 2814–2822. [Online]. Available: http://papers.nips.cc/paper/4878-understanding-dropout.pdf
[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[35] K. Li, Y. Wu, Y. Nan, P. Li, and Y. Li, "Hierarchical multi-class classification in multimodal spacecraft data using DNN and weighted support vector machine," *Neurocomputing*, vol. 259, pp. 55–65, Oct. 2017.
[36] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 1578–1585.
[37] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2017.
[38] B. Bole, C. S. Kulkarni, and M. Daigle, "Adaptation of an electrochemistry-based li-ion battery model to account for deterioration observed under randomized use," SGT, Gurgaon, Haryana, Tech. Rep., 2014.
[39] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 478–487.
[40] U. Shaham, "SpectralNet: Spectral clustering using deep neural networks," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–21. [Online]. Available: https://openreview.net/forum?id=HJ_aoCyRZ

YIXIN LIU received the B.S. degree from Beihang University, Beijing, China, in 2017, where he is currently pursuing the M.S. degree with the Fundamental Science on Ergonomics and Environment Control Laboratory, School of Aeronautics Science and Engineering. His research interests include deep learning (DL), signal processing, and medical image analysis.

KE LI (M'16) received the Ph.D. degree from Beihang University, Beijing, China, in 2008. After two years of postdoctoral research with the School of Aeronautics Science and Engineering, Beihang University, he joined the Fundamental Science on Ergonomics and Environment Control Laboratory, School of Aeronautics Science and Engineering, Beihang University, in April 2010. His current research interests include networked control systems, control methods of thermal engineering, intelligent control algorithms, brain–machine interface, and machine learning.



YUXIANG ZHANG received the B.S. degree from Beihang University, Beijing, China, in 2018, where he is currently pursuing the M.S. degree with the Fundamental Science on Ergonomics and Environment Control Laboratory, School of Aeronautics Science and Engineering. His research interests include deep learning (DL) and natural language processing (NLP).



SHIMIN SONG received the M.S. degree from the School of Astronautic, Beihang University, in 2005. After his graduation, he joined the China Academy of Space Technology, Beijing, China.