



# Application of Multiscale Learning Neural Network Based on CNN in Bearing Fault Diagnosis

Daichao Wang<sup>1</sup> · Qingwen Guo<sup>1</sup> · Yan Song<sup>1</sup> · Shengyao Gao<sup>2</sup> · Yibin Li<sup>1</sup>

Received: 1 February 2019 / Revised: 31 May 2019 / Accepted: 19 June 2019 / Published online: 20 July 2019  
© Springer Science+Business Media, LLC, part of Springer Nature 2019

## Abstract

With the application of intelligent manufacturing becoming more and more widely, the losses caused by mechanical faults of equipment increase. Identifying and troubleshooting faults in an early stage are important. The process of traditional data-driven fault diagnosis method includes data acquisition, fault classification, and feature extraction, in which classification accuracy is directly affected by the result of feature extraction. As a common deep learning method in image recognition, the convolutional neural network (CNN) demonstrates good performance in fault diagnosis. CNN can adaptively extract features from original signals and eliminate the effect of conventional handcrafted features. In this study, a multiscale learning neural network that contains one-dimension (1D) and two-dimension (2D) convolution channels is proposed. The network can learn the local correlation of adjacent and nonadjacent intervals in periodic signals, such as vibration data. The Paderborn data set is came into use to demonstrate the classification accuracy of the method which is brought forward, which includes three conditions of healthy, outer ring (OR) damage and inner ring (IR) damage. The classification accuracy of the method which is put forward is up to 98.58%. The same dataset was applied to test the classification accuracy of support vector machine (SVM) for comparison. And the proposed multiscale learning neural network demonstrates considerable improvements.

**Keywords** Data-driven · Convolutional neural network · Fault diagnosis · Multiscale learning neural network

## 1 Introduction

Recent developments in intelligent production have intensified the need for early fault diagnosis. Early detection of emerging faults is crucial for intelligent production systems because it can prevent unexpected shutdowns and save considerable cost [1–3]. Model-based, knowledge-based (also known as data-driven), and hybrid approaches are considered

the three major categories of fault diagnosis. The acquisition of equipment operation data has become easy because of the development of intelligent manufacturing, and this situation promoted the development of data-driven fault diagnosis methods which require a large amount of historical data but do not need to establish an explicit system model. Therefore, these methods are suitable for complex systems with high integration [4, 5].

With more and more advantages of artificial intelligence (AI), many scholars began to engage in related research. The foundations and algorithms of reinforcement learning (RL) are systematically presented by Li C. et al., and some cyber-physical systems examples are given [6]. Meanwhile the AI is increasingly used in various fields, especially in the industrial fields. Han Qiu et al. proposed a practical data fusion trust system for decision making in the smart vehicle driving systems [7]. The discrete wavelet transform (DWT) was used in cloud servers to protect user data by Qiu H. et al. [8], and the effectiveness of it have been verified. Gai K. et al. proposed an error prevention adjustment algorithm to alert doctors when they make unusual decisions in tele-health domains [9]. Original fault diagnosis method is to judge fault types by experts according to waveform features after converting the

---

✉ Yibin Li  
liyibing@sdu.edu.cn

Daichao Wang  
zhixinnanhuo@yeah.net

Qingwen Guo  
408699043@qq.com

Yan Song  
ysong@sdu.edu.cn

Shengyao Gao  
gaoshengyao@163.com

<sup>1</sup> Shandong University, Qingdao 266237, Shandong, China

<sup>2</sup> China Naval Academy, Beijing 100161, China

vibration signal into frequency domain. This method relies heavily on expert knowledge. However, artificial intelligence can effectively solve this problem.

The vibration signals are most widely used even though they contain a lot of noise. Because of the large amount of noise, the vibration signal usually needs to be feature extracted before used in fault diagnosis [10]. However, feature extraction is complicated and greatly affects the final results. Principal component analysis (PCA) shows good performance in feature selection. For example, PCA has been applied to the selection of important features after decomposing a vibration signal using wavelet packet transform [11]. Although PCA is effective in representing common features of the same type of data samples, it is unsuitable for distinguishing different samples. Therefore, the main role of PCA is to diminish the dimension of the eigenmatrix. As a common machine learning method, SVM demonstrates remarkable performance in breakdown diagnosis. For example, SVM was used for fault diagnosis by Si J. et al., and the good performance of it was proved when there is a small number of samples compared with popular neural network approaches [12]. A number of researchers have proposed the use of SVM in fault diagnosis in combination with signal processing feature extraction methods to improve accuracy. The raw signal was processed and feature extracted by wavelet denoising, and then adopted SVM to fault diagnosis [13, 14]. Despite a good performance in binary classification, SVM suffers from several major drawbacks, such as low efficiency when dealing with large data or multi-classification problems, overfitting, and low training speed [15, 16]. Compared with SVM, artificial neural network (ANN) shows high precision and strong robustness when samples are abundant. In addition, the data preprocessing methods which are applied to extract feature can improve the performance of ANN. After processing by J48 decision tree (DT) [17] and empirical mode decomposition (EMD) accompany with Hilbert Huang transform [18], the features are chosen as the input of ANN. Nevertheless, the performance of traditional feature extraction methods is limited when processing high-dimension or unstable-signal data. ANN also has the defect of low learning efficiency.

Recent developments in deep learning (DL) have stimulated the interest of researchers. DL is an effective means to overcome the aforementioned defects. Deep neural network (DNN) consists of a mass of neural layers. With the advantage of a deep structure, DNN can automatically extract deep features that can represent classified information in the raw signal, thereby effectively avoiding the defect of hand-crafted features [19]. Considering the good performance of DNN methods in feature extraction, many scholars have applied these methods to fault diagnosis, included auto-encoder (AE) and deep belief network (DBN). Jia F. et al. [20] applied the DNN to trouble diagnosis of rolling bearings and had an vital find that the method which is put forward has the

ability to collect potential characteristics from the raw signals which contain a lot of noise. And it is proved that the method which is put forward has the better classification performance than other shallow networks. Given the powerful ability of DNN to collect potential messages from raw data, Xu F. et al. [21] used DNN to collect features from roller bearing oscillation signals and used PCA to diminish the dimension of these characteristics. Fuzzy C-means (FCM) clustering was then used for roller bearing fault identification. Compared with other combinatorial methods like variation mode decomposition (VMD), FCM shows better performance in fault diagnosis. Although deep auto-encoder networks can extract potential features from raw data, it is hard for them to classify the fault and severity. Chen RX. et al. [22] wedded a deep sparse auto-encoder to extract fault severity features and a classifier layer for severity identification; In order to overcome the problem of overfitting, Gaussian yawp were applied to the training samples. The sparse auto-encoder is also employed to extract features and train a classifier for identifying induction motor faults [23]. Different from the research of Chen RX et al., during the training, a regularization means which is known as “dropout” was used to avoid overfitting.

As a effective DL method, Convolutional neural networks (CNNs) are population for classification application, and have been extensively utilized for image recognition and fault diagnosis. Apart from extracting potential features hidden in signals, CNNs can detect local features. Ince T. et al. [24] proposed 1D CNNs based early fault-detection system. The system can propose raw vibration signals, thus avoiding feature extraction. However, the capability of 1D CNNs to detect the local correlation of signals is deficient. Wen L. et al. [25] proposed a method to divert raw time-domain signal data into 2D grayscale used for 2D CNNs to diagnose faults. The method shows good performance and is worth popularizing in fault diagnosis.

Traditional machine learning methods do not have the ability of feature extraction, so it is necessary to manually extract features according to the actual situation, which greatly increases the complexity of fault diagnosis and makes the diagnosis accuracy not high. A multiscale learning neural network of 1D and 2D CNNs is proposed in this study to solve the above problems. This network combines the advantages of two network structures and can extract 1D features in the signal and determine their local correlation. The proposed network is tested on the Paderborn dataset, and the classification accuracy of it is compared with traditional fault diagnosis methods. Comparisons show that the proposed network can effectively extract 1D and 2D features of the raw signal, and its diagnostic accuracy is considerably improved.

The other part of this paper is summarized as follows. Working principle of 1D and 2D CNNs is briefly introduced in Section 2. A multiscale learning neural network of 1D and 2D CNNs is developed in Section 3. The performance of the

method which is put forward is validated on the Paderborn dataset in Section 4. The verdicts and later study directions are presented in Section 5.

## 2 Review of CNNs

Similar to other computational techniques, CNNs are inspired by the image recognition mechanism of the mammal visual cortex. Unlike in global image processing methods, the images from the retina are processed layer by layer in a distributed manner. A set of neural cells directly acts on the input to collect basic characteristics, such as edges. Convolution is the general way of processing images whose function is spatial linear filtering. Local connectivity, weight sharing and shift invariance are the three most important features of convolution, which makes CNNs require minimal pre-processing. Different from other DNN, CNNs adopt a smaller convolution kernel to process the input image, in order to extract features that are small but crucial [26].

CNNs usually comprise convolution, pooling, and full-connection layers. Jianfeng Zhao et al. [27] proposed local feature learning blocks (LFLBs) to collect the local correlations of enter data, which is made up of one convolution layer and one pooling layer. And the working mechanism of CNNs can be summed up as follows: the convolution kernel slides through the entire image with an appropriate stride, which eventually forms a complete feature map to extract local features. The extracted features vary with the different convolution kernel weight matrices used in different convolution layers. The convolution layer is always connected to a sub-sampling layer (such as max-pooling) by a nonlinear mapping function (such as ReLu). The appropriate subsampling layer exerts a good effect on reducing the dimension of the input without losing information. After the connection of multiple LFLBs, the local features extracted are entered into fully connected layers for classification. During the training course to improve the performance of network, All weights are constantly renewed, such as the convolution kernel in different layers.

### 2.1 1D CNN

To integrate feature extraction and fault classification into a network, we use 1D CNNs to act on raw vibration signals directly. The raw vibration signals, which are linearly scaled into the [0, 1] interval, are utilized as the input of 1D CNNs. An LFLB that contains convolution and subsampling layers is shown in Fig. 1. The  $l$ th-layer LFLB of 1D CNNs is described in this figure, in which the red dotted and green lines represent convolution and subsampling operations, respectively [28].

In 1D CNNs, the extracted features of the  $l$ th-LFLB can be calculated as follows:

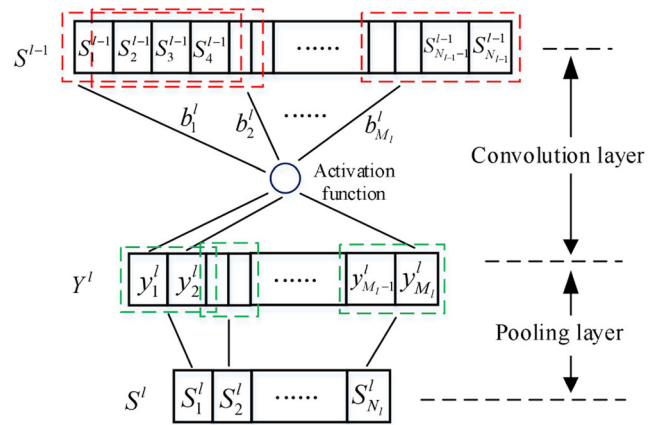


Figure 1 Illustration of the LFLB in 1D CNNs.

$$y_k^l = f \left( b_k^l + \sum_{i=1}^{N_{l-1}} \text{cov } 1D(W_k^l, S_i^{l-1}) \right), \quad k = 1, 2, \dots, M_l \quad (1)$$

where  $N_{l-1}$  represents the quantity of the output at the  $(l-1)$ th-LFLB,  $b_k^l$  represents a scalar bias which is located in the  $k$ th neuron at the  $l$ th-LFLB,  $W_k^l$  is the kernel of the  $k$ th neuron at the  $l$ th-LFLB,  $S_i^{l-1}$  represents the outlet of the  $i$ th neuron at the  $(l-1)$ th-LFLB,  $\text{cov } 1D$  is a 1D convolution operation, and  $f(\cdot)$  represents the activation function of the convolution layer.

A pooling layer follows the convolution layer at the  $l$ th-LFLB, and the output of it can be calculated as follows:

$$S^l = ss(Y^l) \quad (2)$$

where  $ss$  represents the downsampling operation.

The local features that contain implicit fault information are extracted by these LFLBs when the raw vibration signals that are sampled with an appropriate frequency to be a 1D vector are passed to the 1D CNN. The local features learned from raw signal by one-dimensional LFLB are shown in Fig. 2. The input is the amplitude in the time domain, and the colors represent the receptive fields of different 1D LFLB.

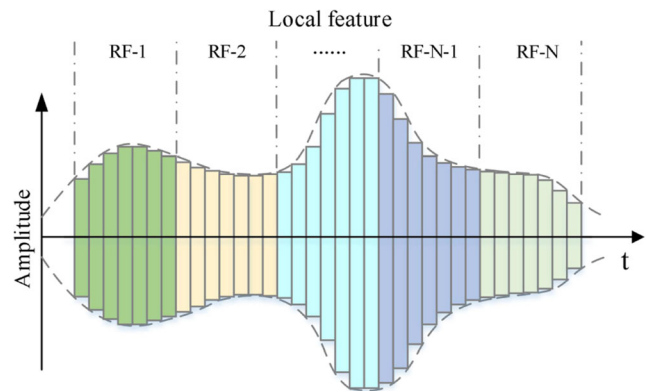


Figure 2 The local features learned from raw signal by one-dimensional LFLB.

The characteristics which is the most prominent of convolution layer are local area connectivity and shared weights, allowing the convolution layer has the function of learning kernel. The convolution layer has a strong capability for local feature extraction due to the convolution operation. The pooling layer have the ability of reducing the dimension of features to make it robust when there is noise or distortion. As the most common subsampling function, max-pooling select the maximum value in the sampling region as the output, which can collect the most important characteristic of the input. The LFLB can be configured in terms of different tasks, and the performance aiming at different tasks varies from different structure and parameters of LFLB.

The local features extracted by these LFLBs are eventually used for fault diagnosis. Softmax is used as the classifier of this architecture, which can be defined as

$$z_i = \sum_j h_j W_{ji} \tag{3}$$

$$\text{softmax}(z_i) = p_i = \frac{e^{z_i}}{\sum_{j=1}^n e^{z_j}} \tag{4}$$

where  $z_i$  and  $h_j$  represent the outlet of penultimate layer and is the inlet of the softmax function.  $W_{ji}$  represents the weight connecting the penultimate and softmax layers.

### 2.2 2D CNNs

Deep convolutional neural networks (DCNNs) have got significant achievements in computer visual sense [29] and are increasingly used in medical imaging [30]. Different from 1D CNNs, 2D CNNs can be used for learning 2D local correlations and extracting hierarchical correlations. The architecture

of typical 2D CNNs, which comprises two stages [31], is shown in Fig. 3. The first stage is composed of a few LFLBs, and the second stage comprises a full connection layer and a classification layer. Similar to an LFLB in 1D CNNs, an LFLB in 2D CNNs contains convolution and pooling layers.

The convolution layers contain a great quantity of convolution kernels, which are also called filters. The input is convoluted by a set of weights to form a feature map. All neurons in the same filter share weights, which cuts the computational complexity of CNNs. If the input of the convolution layer is  $X \in R^{A \times B}$ , in which  $A \times B$  is the dimension of the inlet matrix, then the outlet of the convolution layer is able to be reckoned as follows:

$$C_n = f(X_n * W_n + b_n) \tag{5}$$

where  $X_n$  and  $C_n$  represent the inlet and outlet of the  $n$ th convolution layer, respectively;  $*$  represents the convolution operation;  $W_n$  is a convolution kernel of the  $n$ th convolution layer, and its size is chosen according to the actual situation; and  $b_n$  is the  $n$ th bias. Activation function  $f$  is finally wedded to the consequence.

A pooling layer is always applied after the convolution-layer to decrease the dimension of the input by sub-sampling without loss of useful information. Max, mean, and weighted pooling can be employed in the activation of the pooling layer, but max pooling is the most commonly used in CNNs [32]; it can be calculated as follows:

$$P_n = \max_{C_n \in S} C_n \tag{6}$$

where  $P_n$  represents the outlet of the pooling-layer and  $S$  is the pooling block size. The stride of the pooling-layer is usually

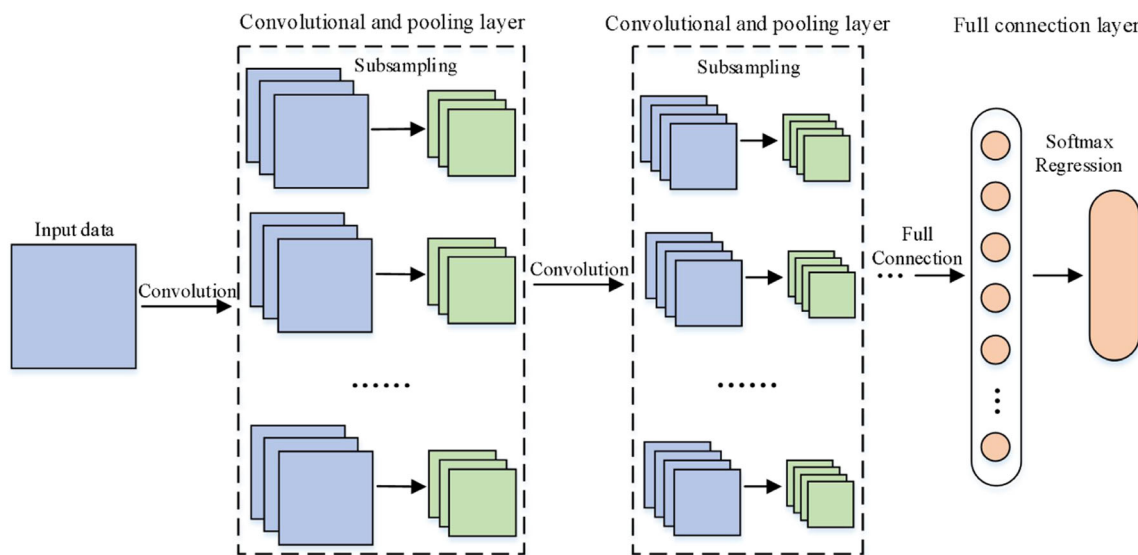


Figure 3 Architecture of typical 2D CNNs.

similar to the length of the pooling window; therefore, the dimension of the output is  $S$  times smaller.

A full-connection layer is always followed by a few LFLBs, which are able to be wedded to different classification models. Softmax regression is normally adopted as the last layer for fault diagnosis, and its output is able to be counted as follows:

$$O = \begin{bmatrix} P(y = 1|x; W_1, b_1) \\ P(y = 2|x; W_2, b_2) \\ \dots \\ P(y = k|x; W_k, b_k) \end{bmatrix} = \frac{1}{\sum_{j=1}^k \exp(W_j x + b_j)} \begin{bmatrix} \exp(W_1 x + b_1) \\ \exp(W_2 x + b_2) \\ \dots \\ \exp(W_k x + b_k) \end{bmatrix} \quad (7)$$

where  $O$  and  $W_j$  are the output of CNN and the weight matrix, respectively.  $b_j$  is the bias.

### 3 Multiscale Learning Neural Network of 1D and 2D CNNs

CNN has been successfully used in numerous applications, such as handwriting, face, and speech recognition. 1D CNN is applied to fault diagnosis due to its good performance in

feature extraction, and 2D CNN is used for detecting 2D features. This indicates that the correlation and information in a local neighbor region are beneficial for fault diagnosis. This section describes the proposed multiscale learning neural network of 1D and 2D CNNs. We applied the min-max normalization method to map the raw vibration signal into a  $[0, 1]$  intermission. The equation is described as follows:

$$x_i^k = \frac{x_i^k - \min(x^k)}{\max(x^k) - \min(x^k)} \quad (8)$$

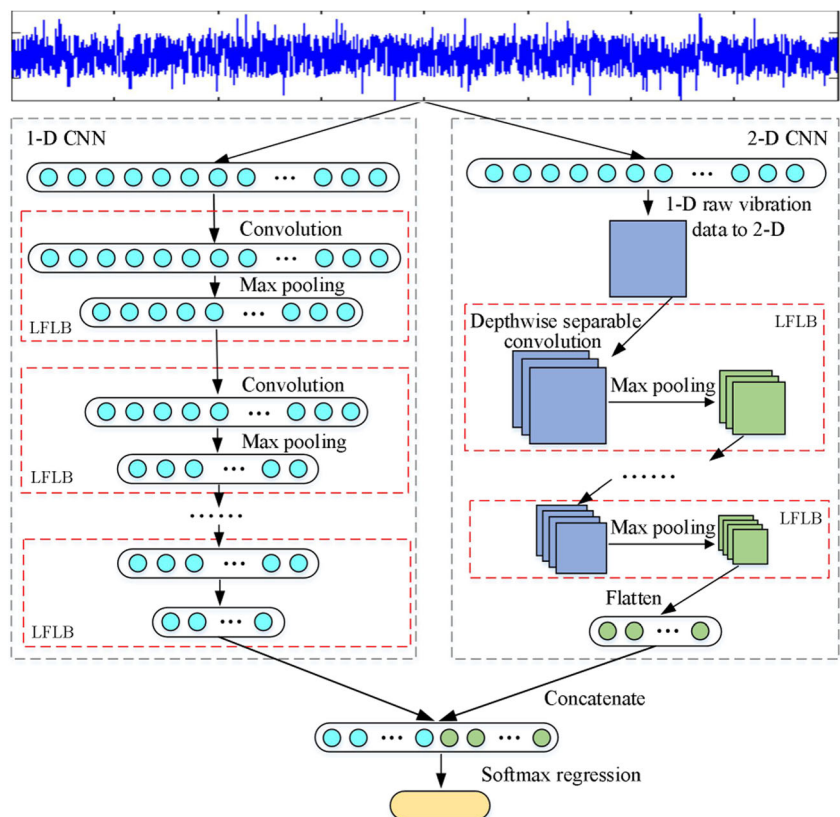
where  $x^k \in R^N$  represents the  $k$ th sample with  $N$  points and  $\min(\cdot)$  and  $\max(\cdot)$  return the minimum and maximum values in sample  $x^k$ .

#### 3.1 Structure of the Proposed Multiscale Learning Neural Network

The proposed 1D and 2D CNNs network structure is shown in Fig. 4. The network contains a one-dimensional CNN and a two-dimensional CNN. The two CNNs are independent of each other in the early stage of feature extraction. After a few LFLBs, the features extracted by 1D and 2D CNNs are fused in the subsequent classification stage and employed for the final fault classification.

For ordinary CNNs, the classification ability is mainly improved by augmenting the quantity of output channels and the

**Figure 4** Structure of the proposed network of 1D and 2D CNNs.



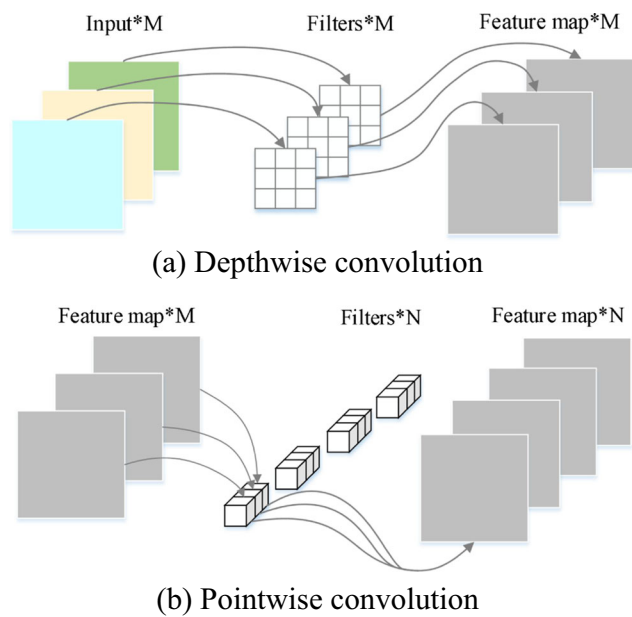
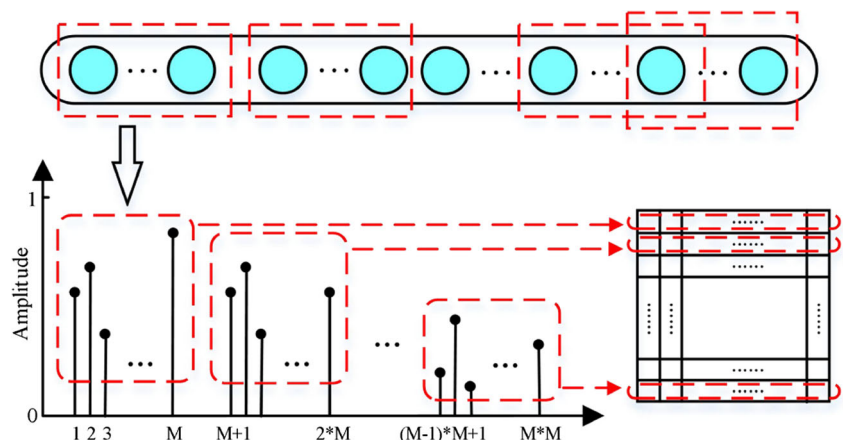


Figure 5 Structure of depthwise separable convolution.

dimension of the convolution kernel; however, this increases the computation and overfitting. The depthwise separable convolution applied in this network can avoid this problem. As shown in Fig. 5, the depthwise separable convolution comprises depthwise and pointwise convolution.

Assuming that we have a convolution layer with kernels of  $D_K \times D_K \times N$ , the input is  $D_F \times D_F \times M$ , the padding is 1, and the stride is 1. The mechanism of depthwise separable convolution is to divide the kernels into depthwise kernels of  $D_K \cdot D_K \cdot M$  and pointwise kernels of  $1 \cdot 1 \cdot N$ . For ordinary CNNs,  $D_K \cdot D_K \cdot M \cdot N \cdot D_F \cdot D_F$  represents the number of operations required, and  $D_K \cdot D_K \cdot N \cdot M$  represents the parameters of the convolution kernel. Meanwhile, for depthwise separable convolution,  $D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + M \cdot N \cdot D_F \cdot D_F$  represents the number of operations required, and  $D_K \cdot D_K \cdot M + N \cdot M$  represents the parameters of the convolution kernel. The results

Figure 6 Illustration of the conversion method for signals from 1D to 2D.



indicate that depthwise separable convolution can reduce parameters without performance degradation.

Once the raw vibration signals have been inputted to the proposed network, the 1D and 2D CNNs are trained simultaneously. For the 1D CNN, the raw vibration data are directly used for network training until the final 1D features are learned, and we assume that these features are  $F_{1-D}$ . However, for the 2D CNN, the raw vibration data must be converted to a 2D image first, which can then be applied to train the network. A full-connection layer follows with a few of LFLBs that is made up of a convolution layer and a max pooling layer to stretch the 2D feature map into 1D data; we assume this data as  $F_{2-D}$ . The final features can be obtained by a concatenate layer, which can be represented as follows:

$$F_f = [F_{1-D}; F_{2-D}] \tag{9}$$

where  $F_f$  is the final features which will be the input of the classifier. Softmax regression is employed as the last layer for classification.

### 3.2 Conversion Method for Signals from 1D to 2D

2D CNN was applied to 2D image recognition and achieved considerable success. Therefore, a conversion method is employed in this study to change the raw vibration signals from 1D to 2D, as shown in Fig. 6.

A possible correlation could exist between adjacent data for a periodic signal. From the Fig. 6 we can see that the raw vibration signals fulfill the 2D matrix by sequence. On the prerequisite of pledging the continuity of the original signal, the non-adjacent data are connected in two dimensions to extract the correlation of the signal in the non-adjacent interval, that is, 2D features. The raw signals are supposed to transform into many sections that contain the same amount of data to obtain the 2D matrix. Considering the diversity of extracted features, the segmentation of sections has the following characteristics:

**Table 1** Operation parameter of healthy bearings during run-in time.

Bearing code	K001	K002	K003	K004	K005	K006
Run-in period (h)	>50	19	1	5	10	16
Radial load (N)	1000–3000	3000	3000	3000	3000	3000
Speed (/min)	1500–2000	2900	3000	3000	3000	2900

- 1) The amount of data in each section is M.
- 2) Sections can be discontinuous.
- 3) Overlapping areas can exist between blocks.

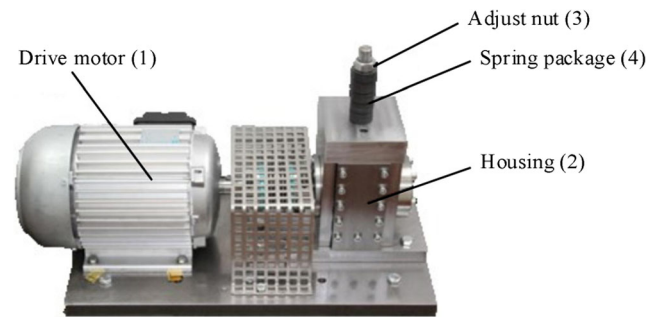
In previous applications,  $2^n$  has often been used as the size of the input image, such as 16, 32, and 64, and has been proven to be effective. In this study, 64 is selected because of the large amount of data in the dataset used in this work. The conversion method gives an valid method to learn 2D features of raw vibration data. This method can also be counted without predefining extra parameters. Therefore, using this method does not increase the amount of calculation in the training process.

### 4 Case Studies and Experimental Results

The Paderborn dataset, which contains healthy bearings and those with OR and IR faults, is wedded to this model to testify the effectiveness of the proposed means. Artificial and real damage are present in OR and IR. The models which is put forward use Python 3.5 with TensorFlow to write and run on Ubuntu 16.04 with a TITAN V GPU.

#### 4.1 Description of the Dataset

In practical applications, due to the long service life of most bearings, the system training dataset for obtaining different damages is complicated; moreover, if damage is identified, then the bearing will be immediately replaced. Consequently, only a small amount of data is available for research. Aiming at this issue, Lessmeier C. et al. [33] provided Paderborn datasets comprising operating data from healthy bearings, bearings with artificial damage, and those with real damages due to an accelerated lifetime test. In order to maintain the controllability of the experiment, we only used ball



**Figure 7** Apparatus for the accelerated lifetime test.

bearings of type 6203 in the research. In Table 1, the descriptions of healthy bearings are listed.

Damages are supposed to be made on bearings to obtain a dataset from damaged bearings. The three different methods for creating artificial damages are electric discharge machining (EDM), drilling, and manual electric sculpting. In Table 2, the details of bearings with artificial damage are listed. As we can see in the table, KA represents the bearing number of outer ring fault while KI represents the bearing number of outer ring fault. IR denotes a bearing with damage of inner race and OR denotes that of outer race. D and E of the damage method represent drilling and electric engraving, respectively.

Real damages on ball bearings are obtained from an accelerated lifetime test rig comprising a bearing housing and an electric motor, as can be seen in Fig. 7. The particulars of bearings with real damage are listed in Table 3, in which F and P in the damage method represent fatigue and plastic deformation, respectively.

After obtaining the damaged bearing, we gathered the dataset from a modular examine device consisting of an electric motor (1), a torque measurement pole (2), a rolling bearing examine module (3), a flywheel (4), and a load motor (5), as shown in Fig. 8. Ball bearings which have different damages are installed in the bearing examine module to produce the data of experiment.

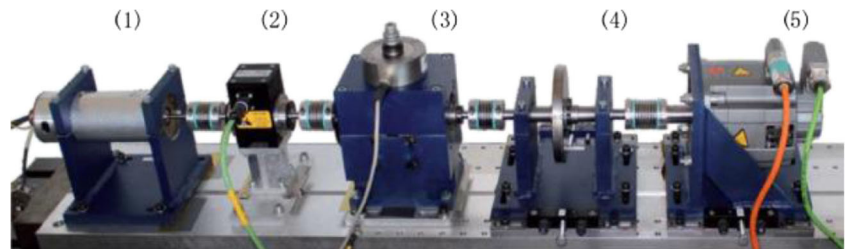
In this experiment, the data collected from K001 and K002 are employed as testing data due to their long running time; others are used as training data. The data gathered from the man-made damaged bearing are also applied as training data, and those collected from the bearings with real damage are applied as testing data.

**Table 2** Bearings with artificial damage.

Bearing code	KA01	KA03	KA05	KA06	KA07	KA09	KI01	KI03	KI05	KI07	KI08
Component	OR	OR	OR	OR	OR	OR	IR	IR	IR	IR	IR
Damage extent	1	2	1	2	1	2	1	1	1	2	2
Damage method	EMD	E	E	E	D	D	EMD	E	E	E	E

**Table 3** Bearings with real damage.

Bearing code	KA15	KA16	KA22	KA30	KI14	KI16	KI17	KI18	KI21
Component	OR	OR	OR	OR	IR	IR	IR	IR	IR
Damage extent	1	2	1	1	1	3	1	2	1
Damage method	P	F	F	P	F	F	F	F	F

**Figure 8** Modular test rig.

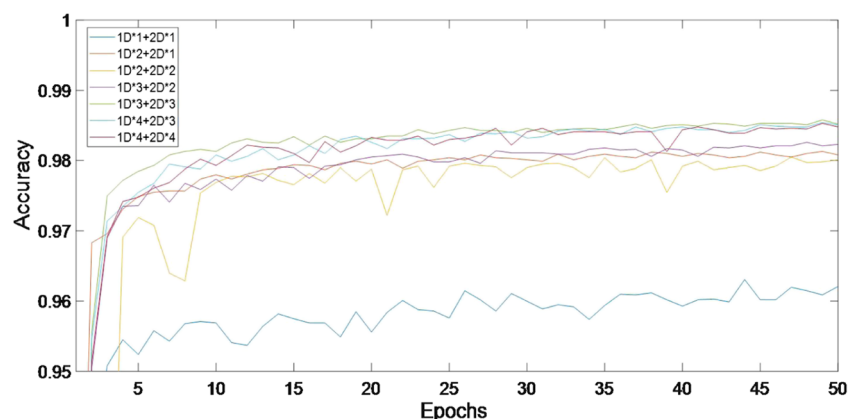
## 4.2 Model Design and Experiment Results

The dataset needs to be processed in the beginning of the experiment. The vibration data in the Paderborn dataset are split into a number of sections, and each section contains 4096 pieces of data. The vibration signal is changed into the frequency domain using a fast Fourier transform (FFT) to upgrade the classification effect of the model. Then, a label is added to the data, in which 0, 1, and 2 represent healthy, OR damage, and IR damage, respectively, without considering artificial and real damages. All healthy data are connected together, and OR and IR damage data are processed similarly. After shuffling the data of the three states, 15,000 pieces of data are selected from each of the three states for training the proposed model. The remaining data are adopted as testing data to testify the classification precision of the model. The model is practiced in 50 epochs. The accuracy of models with different structures is verified to explore the influence of the difference in layers between 1D and 2D convolution on model accuracy, as can be seen in Fig. 9. The parameters of different structures are listed in Table 4. In Table 4 and Fig. 9, 1D\*3 +

2D\*3 denotes the presence of three 1D convolution layers and three 2D convolution layers. The kernel size and strides of the max-pooling layer in 1D convolution are 3, whereas they are both (2,2) in 2D convolution. Consequently, in Table 4, only the parameters of the convolution layer are listed.

1D convolution can achieve a good classification effect, and 2D convolution can find the correlation of signals in non-adjacent intervals that 1D convolution cannot find. In this experiment, network depth is increased layer by layer in the form of 1D convolution having the same number of layers as 2D convolution or 1D convolution having one more layer than 2D convolution to determine the network structure with the highest accuracy of classification. As we can see from Fig. 9, with the deepening of the network layer, classification accuracy is improved until three 1D convolution and three 2D convolution layers remain. The classification accuracy of the model decreases when four 1D and 2D convolution layers exist. Consequently, the model with three 1D and three 2D convolution layers is selected as the best structure.

We conducted a comparative experiment to examine if the network structure which is put forward is able to make the

**Figure 9** Accuracy of models with different structures.

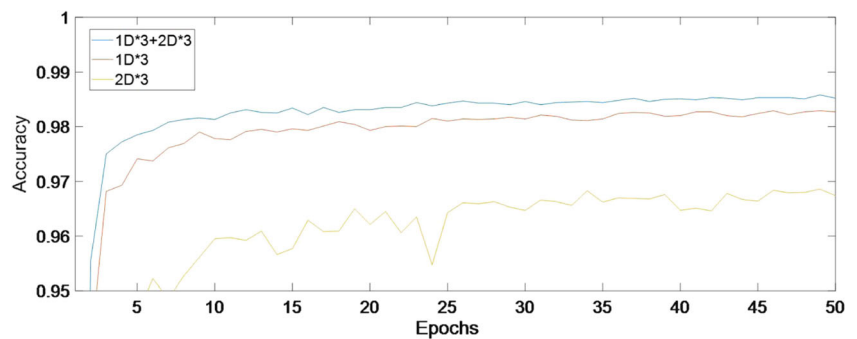


**Table 4** Parameters of different structures in Fig. 9.

Model structure	1D * 1 + 2D * 1	1D * 2 + 2D * 1	1D * 2 + 2D * 2	1D * 3 + 2D * 2	1D * 3 + 2D * 3	1D * 4 + 2D * 3	1D * 4 + 2D * 4
Conv1D Parameters	Layer1 (filters = 16, kernel_size = 7, strides = 1)	Layer1 (filters = 16, kernel_size = 7, strides = 1) Layer2 (filters = 32, kernel_size = 5, strides = 1)	Layer1 (filters = 16, kernel_size = 7, strides = 1) Layer2 (filters = 32, kernel_size = 5, strides = 1)	Layer1 (filters = 16, kernel_size = 7, strides = 1) Layer2 (filters = 32, kernel_size = 5, strides = 1) Layer3 (filters = 64, kernel_size = 5, strides = 1)	Layer1 (filters = 16, kernel_size = 7, strides = 1) Layer2 (filters = 32, kernel_size = 5, strides = 1) Layer3 (filters = 64, kernel_size = 5, strides = 1) Layer4 (filters = 64, kernel_size = 3, strides = 1)	Layer1 (filters = 16, kernel_size = 7, strides = 1) Layer2 (filters = 32, kernel_size = 5, strides = 1) Layer3 (filters = 64, kernel_size = 5, strides = 1) Layer4 (filters = 64, kernel_size = 3, strides = 1)	Layer1 (filters = 16, kernel_size = 7, strides = 1) Layer2 (filters = 32, kernel_size = 5, strides = 1) Layer3 (filters = 64, kernel_size = 5, strides = 1) Layer4 (filters = 64, kernel_size = 3, strides = 1)
Conv2D Parameters	Layer1 (filters = 16, kernel_size = (5,5), strides = (1,1))	Layer1 (filters = 16, kernel_size = (5,5), strides = (1,1))	Layer1 (filters = 16, kernel_size = (5,5), strides = (1,1)) Layer2 (filters = 32, kernel_size = (5,5), strides = (1,1))	Layer1 (filters = 16, kernel_size = (5,5), strides = (1,1)) Layer2 (filters = 32, kernel_size = (5,5), strides = (1,1)) Layer3 (filters = 64, kernel_size = (3,3), strides = (1,1))	Layer1 (filters = 16, kernel_size = (5,5), strides = (1,1)) Layer2 (filters = 32, kernel_size = (5,5), strides = (1,1)) Layer3 (filters = 64, kernel_size = (3,3), strides = (1,1)) Layer4 (filters = 64, kernel_size = (3,3), strides = (1,1))	Layer1 (filters = 16, kernel_size = (5,5), strides = (1,1)) Layer2 (filters = 32, kernel_size = (5,5), strides = (1,1)) Layer3 (filters = 64, kernel_size = (3,3), strides = (1,1)) Layer4 (filters = 64, kernel_size = (3,3), strides = (1,1))	Layer1 (filters = 16, kernel_size = (5,5), strides = (1,1)) Layer2 (filters = 32, kernel_size = (5,5), strides = (1,1)) Layer3 (filters = 64, kernel_size = (3,3), strides = (1,1)) Layer4 (filters = 64, kernel_size = (3,3), strides = (1,1))

**Table 5** Parameters of different structures in Fig. 10.

Model structure	1D * 3 + 2D * 3	1D * 3	2D * 3
Conv1D Parameters	Layer1(filters = 16, kernel_size = 7, strides = 1) Layer2(filters = 32, kernel_size = 5, strides = 1) Layer3(filters = 64, kernel_size = 5, strides = 1)	Layer1(filters = 16, kernel_size = 7, strides = 1) Layer2(filters = 32, kernel_size = 5, strides = 1) Layer3(filters = 64, kernel_size = 5, strides = 1)	*
Conv2D Parameters	Layer1(filters = 16, kernel_size = (5,5), strides = (1,1)) Layer2(filters = 32, kernel_size = (5,5), strides = (1,1)) Layer3(filters = 64, kernel_size = (3,3), strides = (1,1))	*	Layer1(filters = 16, kernel_size = (5,5), strides = (1,1)) Layer2(filters = 32, kernel_size = (5,5), strides = (1,1)) Layer3(filters = 64, kernel_size = (3,3), strides = (1,1))

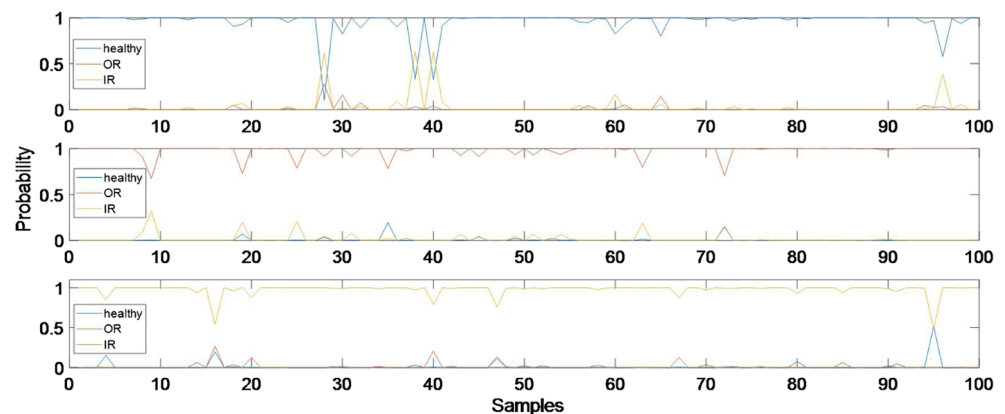
**Figure 10** Accuracy of the proposed network and ordinary convolutional networks.

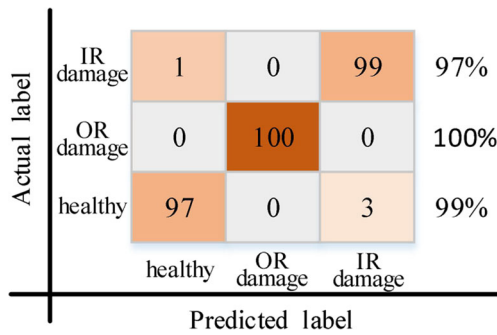
performance of ordinary convolutional networks better, and the parameters of different structures and the results can be seen in Table 5 and Fig. 10.

As we can see in Fig. 10, the performance of one-dimensional CNN is already good, but two-dimensional CNN does not work as well due to its instability and low accuracy, indicating that one-dimension CNN is more suitable than two-dimension CNN for fault diagnosis based on waveform signals. However, the classification accuracy of 1D CNN is improved when combined with 2D CNN. The 2D features are connected to the features extracted by 1D CNN through the concatenated layer in the network. Therefore, 2D CNN is

proven to have the capability to extract several features that 1D CNN cannot, and these can be called 2D features. The classification accuracy of the network is improved because the receptive field is increased after the addition of 2D CNN.

A total of 100 samples are chosen from healthy, OR damage, and IR damage to verify the performance of the best model with three one-dimensional CNN and three two-dimensional CNN and to show the classification effect of the method which is put forward. Figure 11 demonstrates the classification results of healthy, OR damage, and IR damage from top to bottom. Three curves in each subgraph correspond to the probability of the three labels. Similar to the first subgraph, the probability

**Figure 11** Classification results of the best model with three one-dimensional CNN and three two-dimensional CNN.



**Figure 12** Confusion matrix of the best model with three 1D CNN and three 2D CNN.

of the healthy condition is lower than that of IR in three samples, which means that three samples of the healthy condition are misjudged to have IR damage.

The confusion matrix of the best model with three one-dimensional CNN and three two-dimensional CNN is presented in Fig. 12. From the table above we can see that when the classification precision of OR damage is 100%, three samples of the healthy condition are misjudged to have IR damage, and one sample of IR damage is misjudged to be in a healthy condition. This finding means that the proposed model is better at finding OR damage, but errors exist in distinguishing between healthy condition and IR damage.

We drew a comparison between the results of the model which is put forward and the results of the SVM. In Table 6 we are able to see the data that the accuracy for the training data and testing data of the proposed model is both more than 98%. However, the precision for the training data and testing data of SVM is 100% and 41.3%, respectively. The results of SVM appear to exhibit overfitting, but a great deal of training data can be obtained. The best explanation is that SVM is unsuitable for fault identification of the original vibration signal, and its classification performance can only be realized when combined with a feature extraction method. By contrast, the dimension of samples adopted in this study is 4096, which leads to the low training speed, and it is much lower than that of the proposed model.

### 5 Conclusion and Future Work

A multiscale learning neural network is proposed to study the features of frequency data and directly detect bearing faults. In

**Table 6** Comparison of classification accuracy between SVM and the proposed model.

Model	SVM	Proposed model (1D * 3 + 2D * 3)
Accuracy for training data	1	0.9896
Accuracy for testing data	0.413	0.9858

the proposed structure, two channel inputs that correspond to one-dimensional CNN and two-dimensional CNN are applied to study different traits. Owing to the different receptive fields between 1D and 2D convolution, 1D CNN can learn the local correlation of adjacent intervals while 2D CNN can learn the correlation of nonadjacent intervals. The proposed method can connect the extracted features from them and enhance the accuracy of fault diagnosis.

The performance of the multiscale learning neural network which is put forward is tested on the Paderborn dataset. The results show that one-dimensional CNN has better performance than two-dimensional CNN in fault diagnosis when they are used alone. However, when combined with 2D CNN, the classification accuracy of 1D CNN increases to as high as 98.5%, which proves that 2D CNN is equipped to learn the features that 1D CNN cannot. Furthermore, the same data structure with the model which is put forward is employed for SVM, and the consequence indicated that the performance of multiscale learning neural network which is brought forward is much better than that of SVM.

However, although the proposed multiscale learning neural network is proven to be suitable for fault diagnosis of vibration signals, it may not demonstrate the same performance in the classification of other signals. In the future, other structures will be considered to increase the generalization capability of the model. We will also make research of transfer learning to shorten the training time of the model.

### References

- Dai, X., & Gao, Z. (2013). From model, signal to knowledge: A data-driven perspective of fault detection and diagnosis. *IEEE Transactions on Industrial Informatics*, 9(4), 2226–2238.
- Qin, F. W., Bai, J., & Yuan, W. Q. (2017). Research on intelligent fault diagnosis of mechanical equipment based on sparse deep neural networks. *J. Vibroeng.*, 19, 2439–2455 [CrossRef].
- Brkovic, A., Gajic, D., Gligorijevic, J., et al. (2017). Early fault detection and diagnosis in bearings for more efficient operation of rotating machinery. *Energy*, 136, 63–71.
- Gou, X., Bian, C., Zeng, F., et al. (2018). A Data-Driven Smart Fault Diagnosis Method for Electric Motor. 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C). IEEE, 250–257.
- Wen, L., Gao, L., & Li, X., et al. (2017). A new data-driven intelligent fault diagnosis by using convolutional neural network. 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM). IEEE, 813–817.
- Li, C., & Qiu, M. (2019). *Reinforcement Learning for Cyber-Physical Systems: with Cybersecurity Case Studies*. Boca Raton: Chapman and Hall/CRC.
- Han, Q. I. U., Meikang, Q. I. U., Zhihui, L. U., et al. (2019). An efficient key distribution system for data fusion in V2X heterogeneous networks. *Information Fusion*, 50, 212–220.
- Qiu, H., Noura, H., Qiu, M., et al. A User-Centric Data Protection Method for Cloud Storage Based on Invertible DWT. *IEEE Transactions on Cloud Computing*, PP(99), 1–1.

9. Gai, K., Qiu, M., & Chen, L. C., et al. (2015). Electronic Health Record Error Prevention Approach Using Ontology in Big Data. IEEE International Conference on High Performance Computing & Communications. IEEE.
10. Yang, T., Guo, Y., Wu, X., et al. (2018). Fault feature extraction based on combination of envelope order tracking and cICA for rolling element bearings. *Mechanical Systems and Signal Processing*, *113*, 131–144.
11. Agrawal, S., Giri, V. K., & Tiwari, A. N. (2018). Induction motor bearing fault classification using WPT, PCA and DSVM. *Journal of Intelligent & Fuzzy Systems*, (Preprint): 1–12.
12. Si, J., Li, Y., & Ma, S. (2018). Intelligent fault diagnosis for industrial big data. *Journal of Signal Processing Systems*, *90*(8–9), 1221–1233.
13. Konar, P., & Chattopadhyay, P. (2011). Bearing fault detection of induction motor using wavelet and Support Vector Machines (SVMs). *Applied Soft Computing*, *11*(6), 4203–4211.
14. Abbasion, S., Rafsanjani, A., Farshidianfar, A., et al. (2007). Rolling element bearings multi-fault classification based on the wavelet denoising and support vector machine. *Mechanical Systems and Signal Processing*, *21*(7), 2933–2945.
15. Santos, P., Villa, L. F., Reñones, A., et al. (2015). An SVM-based solution for fault detection in wind turbines. *Sensors*, *15*(3), 5627–5648.
16. Xiao, J., & Zhuo, R. (2018). Research on Motor Rolling Bearing Fault Classification Method Based on CEEMDAN and GWO-SVM. 2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). IEEE, pp. 1123–1129.
17. Malik, H., & Yadav, A. (2017). EMD and ANN based intelligent model for bearing fault diagnosis. *Journal of Intelligent & Fuzzy Systems* (Preprint): 1–12.
18. THAMBA N, B., ARAVIND, A., RAKESH, A., et al. (2018). Application of EMD, ANN and DNN for Self-Aligning Bearing Fault Diagnosis. *Archives of Acoustics*, *43*(2), 163–175.
19. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436.
20. Jia, F., Lei, Y., Lin, J., et al. (2016). Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mechanical Systems and Signal Processing*, *72*, 303–315.
21. Xu, F., Wang, D., & Tsui, K. L. (2018). Combining DBN and FCM for Fault Diagnosis of Roller Element Bearings without Using Data Labels. *Shock and Vibration*, 2018.
22. Chen, R., Chen, S., He, M., et al. (2017). Rolling bearing fault severity identification using deep sparse auto-encoder network with noise added sample expansion. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, *231*(6), 666–679.
23. Sun, W., Shao, S., Zhao, R., et al. (2016). A sparse auto-encoder-based deep neural network approach for induction motor faults classification. *Measurement*, *89*, 171–178.
24. Ince, T., Kiranyaz, S., Eren, L., et al. (2016). Real-Time Motor Fault Detection by 1-D Convolutional Neural Networks. *IEEE Transactions on Industrial Electronics*, *63*(11), 7067–7075.
25. Wen, L., Li, X., Gao, L., et al. (2018). A New Convolutional Neural Network-Based Data-Driven Fault Diagnosis Method. *IEEE Transactions on Industrial Electronics*, *65*(7), 5990–5998.
26. Mukhopadhyay R, Panigrahy P S, Misra G, et al. Quasi 1D CNN-based Fault Diagnosis of Induction Motor Drives. 2018 5th International Conference on Electric Power and Energy Conversion Systems (EPECS). IEEE, 2018: 1–5.
27. Zhao, J., Mao, X., & Chen, L. (2019). Speech emotion recognition using deep 1D & 2D CNN LSTM networks. *Biomedical Signal Processing and Control*, *47*, 312–323.
28. Kiranyaz, S., Ince, T., & Gabbouj, M. (2016). Real-time patient-specific ECG classification by 1-D convolutional neural networks. *IEEE Transactions on Biomedical Engineering*, *63*(3), 664–675.
29. Krizhevsky A, Sutskever I, Hinton G E (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*. 1097–1105.
30. Brosch, T., Yoo, Y., Tang, L. Y. W., et al. (2015). Deep convolutional encoder networks for multiple sclerosis lesion segmentation. International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Cham, pp. 3–11.
31. Jing, L., Zhao, M., Li, P., et al. (2017). A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox. *Measurement*, *111*, 1–10.
32. Abdel-Hamid, O., Deng, L., & Yu, D. (2013). Exploring convolutional neural network structures and optimization techniques for speech recognition. *Interspeech*, 2013, 1173–1175.
33. Lessmeier, C., Kimotho, J. K., Zimmer, D., et al. (2016). Condition Monitoring of Bearing Damage in Electromechanical Drive Systems by Using Motor Current Signals of Electric Motors: A Benchmark Data Set for Data-Driven Classification. Proceedings of the European Conference of the Prognostics and Health Management Society.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Daichao Wang** received the master's degrees from College of Automation Engineering, Qingdao University of Science and Technology 2018. He is studying for Ph.D in Institute of Marine Science and Technology (IMST), Shandong University. His research interests include fault diagnosis, and artificial intelligence.



**Qingwen Guo** received his M.Sc. degree in college of Information Science and Engineering, Ocean University of China in 2017. He is now studying for Ph.D. degree in Institute of Marine Science and Technology (IMST), Shandong University (Qingdao) China. His research interests are machinery fault diagnosis, artificial intelligence and deep learning applications.



**Yan Song** received the B.S. and Ph.D. degrees in college of Information Science and Engineering, Ocean University of China in 2012 and 2018, respectively. She is currently an Associate Professor in Institute of Marine Science and Technology (IMST), Shandong University. Her research interests include machine learning application, deep learning and mechanical fault diagnosis.



**Yibin Li** received his PhD from the Department of School of Electronic, Electrical and Systems Engineering, Loughborough University, England, in 2009. During his PhD writing-up stage, he worked in The Institute of Electronics, Communications and Information Technology (ECIT), QUB, North Ireland. In 2010, he joined the embedded group in Computer science & Engineering Department of Shandong University as associate professor.

He has been engaged in several projects including embedded design for Lidar system, the reconfigurable hardware implementation of AES. etc.



**Shengyao Gao** received his PhD in 2009, and now works for the Naval Research Academy. At present, he is mainly engaged in the research of the ship vibration and noise control.