Mathematical Biosciences and Engineering

Research article

Few-shot bearing fault detection based on multi-dimensional convolution and attention mechanism

Yingying Xu[1,2,3,4]**, Chunhe Song**[1,2,3,*]**and Chu Wang**[1,2,3,*]

1 State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China

2 Key Laboratory of Networked Control Systems, Chinese Academy of Sciences, Shenyang 110016, China

3 Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China

4 University of Chinese Academy of Sciences, Beijing 100049, China

* **Correspondence:** Email: songchunhe@sia.cn, wangchu@sia.cn.

Abstract: Bearings are critical components of industrial equipment and have a significant impact on the safety of industrial physical systems. Their failure may lead to equipment shutdown and accidents, posing a significant risk to production safety. However, it is difficult to obtain a large amount of bearing fault data in practice, which makes the problem of small sample size a major challenge for bearing fault detection. In addition, some methods may overlook important features in bearing vibration signals, leading to insufficient detection capabilities. To address the challenges in bearing fault detection, this paper proposed a few sample learning methods based on the multidimensional convolution and attention mechanism. First, a multichannel preprocessing method was designed to more effectively utilize the information in the bearing vibration signal. Second, by extracting multidimensional features and enhancing the attention to important features through multidimensional convolution operations and attention mechanisms, the feature extraction ability of the network was improved. Furthermore, nonlinear mapping of feature vectors into the metric space to calculate distance can better measure the similarity between samples, thereby improving the accuracy of bearing fault detection and providing important guarantees for the safe operation of industrial systems. Extensive experiments have shown that the proposed method has good fault detection performance under small sample conditions, which is beneficial for reducing machine downtime and economic losses.

Keywords: bearing fault detection; few-shot learning; deep learning; nonlinear mapping

1. Introduction

In industrial systems, bearings are important mechanical components, and their normal operation is crucial to ensure the stability and safety of the system [1]. As a key mechanical component, bearing failure can seriously affect the reliable operation of the system. Under long-time operation and harsh environments, bearings are prone to failure, leading to system performance degradation and even accidents. The bearing fault detection has been facing technical difficulties such as complex signals, diverse fault modes, and weak early fault characteristics, which is a hot spot in this research field. It has been found that the probability of bearing failure is the highest among other components [2], and more than 41% of machine failures are caused by bearings [3]. Therefore, bearing failure detection is important for industrial systems. Bearings in different operating conditions have different levels of vibration and noise, and these bearing vibration signals reflect the mechanical operation in real time [4]. In addition, the rapid development of sensor technology makes the acquisition of vibration signals more convenient. Therefore, the acquisition and analysis of vibration signals is a commonly used rolling bearing fault diagnosis method [5].

Traditionally, the research on bearing fault detection is mainly focused on the field of signal analysis, which is mainly to obtain the time-domain, frequency-domain and time-frequency characteristics of the vibration signals, and the commonly used methods are power spectrum analysis [6], cepstrum analysis [7], envelope spectral analysis [8], wavelet analysis [9], continuous wavelet transform (CWT) [10], and empirical modal decomposition (EMD) [11]. Although they have achieved some success, these methods rely on manually designed features and have weak generalization ability, making them difficult to apply to new scenarios. Deep learning is an effective solution for this issue. Deep learning allows for layer-by-layer feature extraction through multilayer neural networks, which can automatically learn to represent features in the data [12, 13]. This enables deep learning for fault detection in vibration signals to better capture complex features in the signals. Currently, there are many deep learning based methods, such as recurrent neural networks (RNN) [14], long short-term memory networks (LSTM) [15], convolutional neural networks (CNN) [16], deep ensemble learning network [17], multi-attention fusion residual convolutional neural network [18], deep convolutional variational autoencoder [19], etc. Through ingeniously integrating multilayer network architectures and complex data representation capabilities,these methods have significantly propelled innovation and progress in the field of fault detection and feature extraction in deep learning.

However, most of the deep learning based methods rely on a large amount of sample data for training, but in practice, this poses a challenge for fault detection due to the variation of bearing vibration information under different operating conditions and the difficulty of obtaining a large amount of sample data. Therefore, how to achieve accurate fault detection using few samples has become a hot topic. At present, small sample learning methods mainly include data augmentation based methods, meta learning based methods, transfer learning based methods, metric learning based methods, etc. Data augmentation is a method of addressing insufficient sample size by directly increasing the diversity of sample size and distribution [20–23]. Data augmentation can be combined with other methods to improve detection performance under small samples. However, due to the insufficient number of annotated samples, simply enhancing the sample and feature space for small samples can only bring limited performance improvement, making it difficult to fundamentally solve the problem of small sample object detection. Meta learning is to transfer prior knowledge from annotated source domains to new

domains with few data by simulating a series of similar small sample training tasks [24]. It can quickly update model parameters with a small number of support set samples with only a few iterations under specific tasks. However, meta learning requires manually constructing the support set in the task and can only perform pretraining and transfer on fixed tasks. Furthermore, it usually has high computational complexity and prunes to non-convergence issues during the learning iteration process. Metric learning maps the features of potential targets and basic data to the same embedding space, then classifies them through similarity measurement [25]. Metric learning generally needs to solve the following three problems: class prototype representation of base classes, measurement mechanism of bounding boxes, and loss function design. Metric learning is easy to implement incremental learning because after training the model on the base class dataset, it can be directly used to detect new classes. However, when the data volume is large and the feature dimension is high, metric learning has problems such as long computation time and high memory consumption, which reduces the real-time performance of the algorithm. Transfer learning is also to transfer prior knowledge from annotated source domains to new domains with little data. Typically, transfer learning methods include fine tuning, multitask learning, domain adverse training, zero shot learning, etc. Compared with meta learning, transfer learning based methods do not require designing small samples training tasks, making them widely used. Currently, some works [26–29] have attempted to use transfer learning to address the issue of small sample fault detection. However, transfer learning still faces some challenges and difficulties, such as establishing a correspondence between the source domain and the target domain and maintaining the performance of the source domain. These methods mentioned above do not begin from fully explore the hidden information of the data itself; therefore, this paper aims to solve the problem of fewer samples in bearing fault detection by using the characteristics of the finite original signal itself to improve the efficiency and performance of small-sample learning.

To overcome the above issues, this paper proposes a few shot bearing fault detection methods. The main contributions of this paper include:

1) A bearing fault detection model using multidimensional convolution and attention is proposed, which adapts to few-sample conditions via a tailored network structure.

2) A data conversion module is designed to form multichannels by combining various data preprocessing methods, which effectively retains key edge information and fully utilizes data information.

3) A feature extraction module is designed, which combines self-attention mechanism with multi-scale CNNs , enabling a more comprehensive capture of data features and an improved performance of the proposed method.

4) A sample similarity measurement module is designed, which maps features to a measurement space for similarity assessment and effectively distinguishes intrinsic data differences to enhance the network's ability to measure sample similarity.

The rest of the paper is as follows. Section 2 describes the related work, Section 3 describes the framework of the proposed method and the details of each modular part, Section 4 gives the experimental results, and Section 5 gives the conclusion.

2. Related work

Bearing fault detection is a typical classification and anomaly detection problem, often using the vibration signal to determine its operating state. The key problem of bearing fault detection is how

to accurately extract features from the sensor data and design an effective model. Meanwhile, it also needs to consider the adaptability and robustness in the actual working conditions.

2.1. Bearing fault detection methods based on traditional methods

Traditional bearing fault detection methods mainly rely on characterizing the signal in the time, frequency, and time-frequency, domains to extract features, and using classifiers to discriminate different fault types. Li et al. [30] effectively extracted the characteristic frequencies of inner and outer ring faults by proposing an adaptive morphological update to enhance the wavelet transform. Fu et al. [31] considered the nonlinear non-Gaussian non-smooth features of the signal, and used ensemble empirical mode decomposition (EEMD) for decomposition of the original signal. This method extracts the root mean square value and power spectrum center of mass features as inputs and uses the optimized Elman AdaBoost model for classification and identification of bearing faults. Zheng et al. [32] proposed an adaptive power spectrum Fourier decomposition method to solve the problems of too many components and cross-mixing in Fourier decomposition. This method works by automatically searching the intervals of each component in the power spectrum of the original signal and decomposing the signal into multiple single components. The fault feature contained in these single components can be used to diagnose bearing faults. Konar and Chattopadhyay [10] considered the unsuitability of Fourier analysis for analyzing nonstationary and transient signals and proposed the use of CWT for feature extraction, then input the features into support vector machines (SVMs) to detect the bearing faults in induction motors. These methods based on signal analysis have been widely used; however, they suffered from the limitations of relying on expertise and a prior experience, as well as the need to manually design the feature extraction.

2.2. Deep learning based bearing fault detection method

With the continuous development and improvement of deep learning technology, deep learning-based bearing fault detection methods play a more important role in future industrial applications. Deep learning-based methods can learn complex feature representations from large amounts of raw sensor data without relying on expertise and manual feature engineering to address bearing fault detection. Deep learning has been applied in emerging areas of industry such as state feature extraction methods using similarity to monitor the propagation process of gear surface wear [33] and using digital twins to solve the problem of monitoring and evaluating surface wear in industrial gear systems [34]; these studies provide more possibilities for deep learning in industrial applications. In addition, to address the needs of bearing fault detection, Ni et al. [35] proposed a deep learning network structure, pulse-Coupled integrated residual network (PIResNet), to solve the problem of fault diagnosis of rolling bearings under different operating conditions, using the method of deep learning of physical information. Peng et al. [36] proposed a deeper one-dimensional convolutional neural network (Der-1DCNN) deep convolutional neural network method based on 1D residual blocks to address the needs of high-speed train bearings for fault detection in strong noise environments and variable load conditions. The one-dimensional convolutional approach can capture local temporal features in the signals, but it is unable to capture long-term dependencies due to the limited coverage of convolutional kernels. Peng et al. [37] converted one-dimensional time series signals into two-dimensional image signals as inputs to a two-dimensional convolutional neural network (2D-CNN) model through a linear mapping

for bearing fault identification and classification. However, this method does not take into account the problem of incoherent edge information in each row in the 2D convolution. 2D convolution can enable the model to capture the periodic changes on different time scales in the data at the same time, but its sensory field is fixed, which may not be able to flexibly adapt to the dependencies of different time spans. Yu et al. [15] proposed a hierarchical algorithm based on stacked LSTM networks for bearing fault diagnosis, which directly takes the raw timing signals as inputs and extracts the features automatically. LSTM can maintain and update the information in long sequences by the design of the gating structure and the memory cells to deal with the long-term temporal relationships. However, due to the fixed length of the memory cells, it is often only able to capture local dependencies rather than global dependencies of the entire sequence. Some works have proposed combining the above methods with each other. Combining the above methods can take advantage of their respective strengths to improve the performance of the model in feature extraction. For example, Wang et al. [38] automatically learned the features of the signal at different scales by combining two channels, 1D CNN and 2D CNN, and the network can learn the local correlation between neighboring and non-neighboring intervals of the periodic signal. Khorram et al. [39] proposed an end-to-end 1D CNN+LSTM network architecture that considers both local and global features of time series.

2.3. Solutions for few samples

Although deep learning based methods have achieved great success, most of these methods rely on a large amount of data for training and optimization. However, most of the data is normal with only a small proportion of faulty data, which is a typical few-sample problem. Therefore, some works have proposed solutions to the few-sample problem. Liu et al. [20] used a generative adversarial networks (GAN) to construct a generator to obtain reconstructed residuals and enhance the feature extraction capability of the recognizer through an adversarial mechanism. A LSTM based an Autoencode framework is established to reduce the dimensionality of the original sensing data and extract critical time fault features. Yang et al. [21] used a conditional generative adversarial network to learn the distribution of the original 1D data, generated new sample data to expand the sample size, and used 2D-CNN to extract image features and classify bearing fault types. Li et al. [29] used a transfer learning approach utilizing CNNs and multilayer perceptrons (MLPs) as base models, with some of the base models transferred to the target domain for fine-tuning. The above methods for a few samples provide some solutions, but they usually require a large amount of non-primitive data to train and optimize the model. GAN-based methods still require sufficient data to train a stable generator, with limitations such as the quality of generated samples being difficult to assess, and discrepancies between expanded and real data. Transferring-based methods require both source and target domains. As a result, lots of works are needed to ensure effective knowledge transfer between source and target domains. In the case of limited data volume, these methods may not achieve the expected performance.

3. The proposed method

In this paper, a multichannel multidimensional bearing fault detection method is proposed, and the main architecture of the proposed model is shown in Figure 1, where multiple channels are generated by preprocessing the input data through the preprocessing block, then the processed data is input into the model. After the feature extraction block consisting of the multidimensional convolution and the
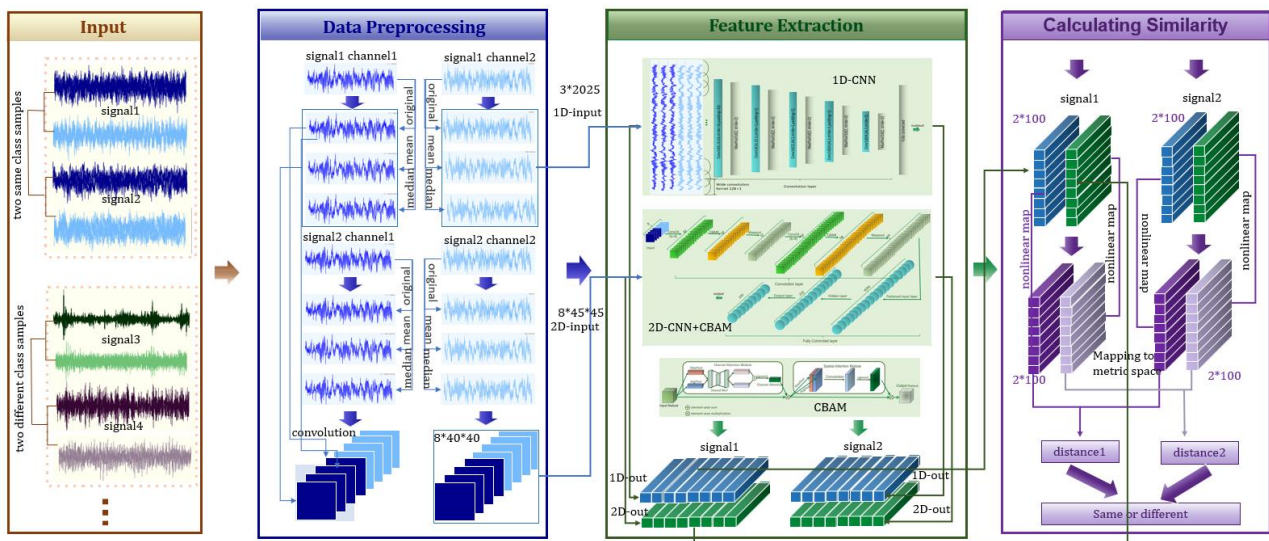
Figure 1. Overall model architecture.

attention mechanism, the extracted feature vectors are passed through the similarity measure block to measure the similarity between the two samples, then the probability that whether two samples belong to the same category is output. The data preprocessing block converts the data into multiple channels through median filtering, mean filtering, and convolution operations, which can retain key edge information and make full use of the information in the data. The feature extraction block captures the data features more comprehensively through multidimensional convolution with hybrid attention mechanism. The similarity metric block maps the extracted feature vectors to the space of the metric through a nonlinear method, then calculates the similarity between the samples. The ability to model the nonlinear relationship between two samples can be improved through the nonlinear mapping, which in turn improves the accuracy of measuring the similarity between samples. The rest of this section describes the data preprocessing, feature extraction, and similarity measure blocks in detail.

3.1. Multichannel data preprocessing

For faulty bearings, abrupt changes in the bearing signal amplitude occur, as the rolling element passes over the faulty region of the bearing. These sudden changes would disturb the overall distribution of the signal and, therefore, can be used as important clues for detecting faulty bearings. In order to fully explore and utilize the information contained in the signal, additional channels processed by median filtering and mean filtering are introduced to the original signal. Adding multiple channels can provide more information to the CNN. In addition, we introduce using 1D convolution as a channel in the process of 2D data conversion. This multichannel fusion preprocessing strategy aims to enhance the model's ability to identify meaningful patterns in the signal while reducing the interference of noise on the analysis results, thus improving the accuracy and reliability of the subsequent analysis. The overall result of this section is shown in the Figure 2.
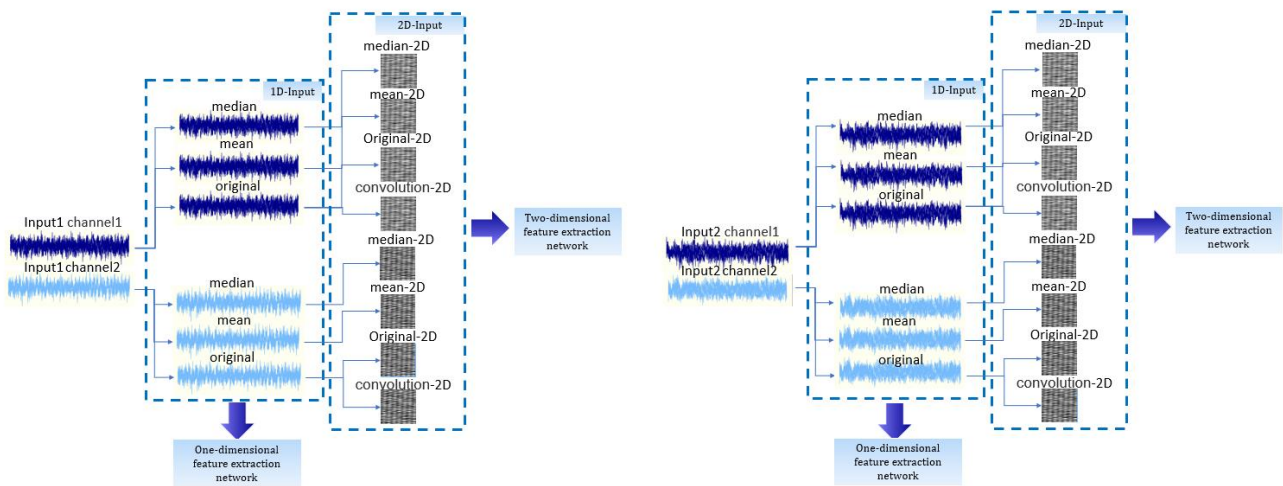
Figure 2. Visual results of multichannel data processing.

3.1.1. Preprocessing of 1D data

Median filtering aims to suppress extreme values and impulse noise, while mean filtering helps to smooth the signal and reduce random fluctuations. Considering the limitations of the subsequent 2D-CNN, the original signal is cropped to the size of N^2, the channels processed by median filtering and mean filtering are additionally introduced on top of the original signal, and the combination of the original data, and the filtered data can provide more information.

The input original signal is X_i, and the output of the median filter with a window number of $2m + 1$ at the t-th data is:

$$X_{\text{imedian}}(t) = \text{Median}\{X_i(t - m), \ldots, X_i(t), \ldots X_i(t + m)\} \tag{3.1}$$

where Median{} denotes the median of all samples taken within the window of processed data at this time. The input original signal is X_i, and the output of the mean filter with a window of $2n + 1$ at the t-th data is:

$$X_{\text{imean}}(t) = \text{Mean}\{X_i(t - n), \ldots, X_i(t), \ldots X_i(t + n)\} \tag{3.2}$$

where Med{} denotes the mean of all samples taken within the window of processed data at this time.

The output after data preprocessing is:

$$X_{i\_1}' = [X_i, X_{\text{imean}}, X_{\text{imedian}}] \tag{3.3}$$

3.1.2. Converting 1D data to 2D data

In order to make the edge information more coherent for the process of 2D data conversion, we introduced 1D convolution as another processing channel on the basis of the above processing method. To begin, the one-dimensional original data X_i is subjected to the convolution operation with kernel K. The result after convolution is:

$$X_{\text{iconv}} = K \otimes X_i \tag{3.4}$$

where $\otimes$ is the convolution operation.

We map the signal from 0 to 255,

$$
\begin{cases}
X_i{}' = g\left(\dfrac{X_i(c) - X_{i\_min}(c)}{X_{i\_max}(c) - X_{i\_min}(c)}\right) \times 255 \\[2ex]
X_{\mathrm{imean}}{}' = g\left(\dfrac{X_{\mathrm{imean}}(c) - X_{\mathrm{imean\_min}}(c)}{X_{\mathrm{imean\_max}}(c) - X_{\mathrm{imean\_min}}(c)}\right) \times 255 \\[2ex]
X_{\mathrm{imedian}}{}' = g\left(\dfrac{X_{\mathrm{imedian}}(c) - X_{\mathrm{imedian\_min}}(c)}{X_{\mathrm{imedian\_max}}(c) - X_{\mathrm{imedian\_min}}(c)}\right) \times 255 \\[2ex]
X_{\mathrm{iconv}}{}' = g\left(\dfrac{X_{\mathrm{iconv}}(c) - X_{\mathrm{iconv\_min}}(c)}{X_{\mathrm{iconv\_max}}(c) - X_{\mathrm{iconv\_min}}(c)}\right) \times 255
\end{cases}
\tag{3.5}
$$

where g( ) means rounding the normalized signal value.

The result of the convolution is added to $X_{i\_1}{}'$ as the channel, which becomes X_{i_2}.

$$
X_{i\_2}{}' = \left[X_i{}', X_{\mathrm{imean}}{}', X_{\mathrm{imedian}}{}', X_{\mathrm{iconv}}{}'\right]
\tag{3.6}
$$

Then the one-dimensional vector X_{i_2} is converted into the desired $4 \times N \times N$ matrix $X_{i\_2}{}'$, denoted as:

$$
\begin{bmatrix}
X_{i\_2}{}'(c) & X_{i_2}{}'(c+1) & \cdots & X_{i\_2}{}'(c+N-1) \\
X_{i\_2}{}'(c+N) & X_{i\_2}{}'(c+N+1) & \cdots & X_{i\_2}{}'(c+2N-1) \\
\vdots & \vdots & \ddots & \vdots \\
X_{i\_}{}'\left(c+N^2-N\right) & \cdots & \cdots & X_{i_2 2}{}'\left(c+N^2-1\right)
\end{bmatrix}
\tag{3.7}
$$

3.2. Multidimensional convolutional and attentions feature extraction block

In this study, a feature extraction block incorporating 1D-CNN, 2D-CNN, and a convolutional block attention module (CBAM) is proposed for bearing fault detection. The block aims to make full use of the feature information in the bearing vibration signals, which can capture the data features more comprehensively and improve the feature extraction capability and model performance.

3.2.1. 1D feature extraction

The 1D convolutional kernel moves along the time axis to capture the local temporal dependencies within the signal. The 1D-CNN utilizes the model in wavelet decomposition CNN (WDCNN) to extract 1D features from the input vibration signals. This model effectively captures the input vibration signals by employing wide convolutional layers and multistage convolutional layers. The use of multilayer convolutional kernels enables the network to delve deeper and extract a well-represented model structure. Through the multilayer convolutional kernels and pooling operations of the WD-CNN model, the temporal features of the vibration signal can be extracted more deeply, resulting in a stronger feature representation. The structure of the WDCNN model is shown in Figure 3:
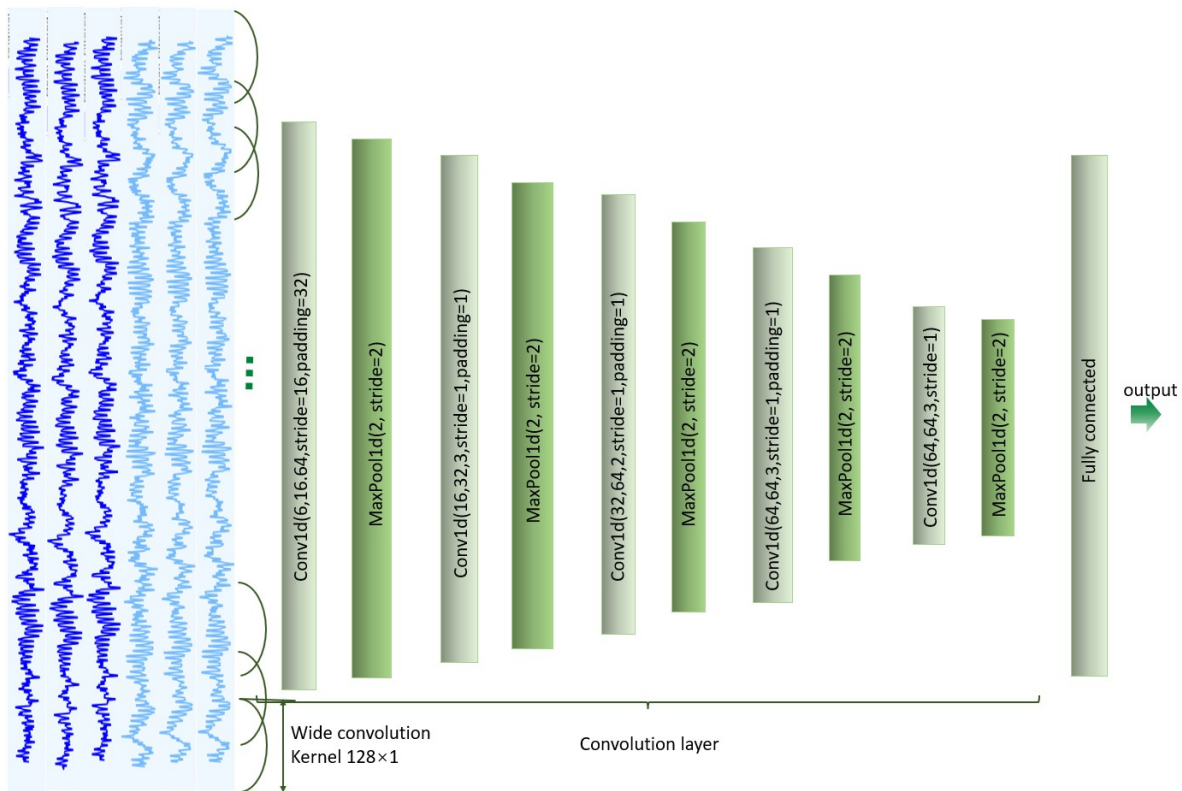
Figure 3. WDCNN structure.

3.2.2. 2D feature extraction

The main structure of 2D feature extraction is designed as shown in Figure 4, which consists of a series of convolutional layers, a CBAM module, and a fully connected layer. 2D-CNN enables the model to capture both spatial information and local features in the data with attention to periodic variations on different time scales. CBAM is introduced to operate channel attention and spatial attention on the features, which can selectively enhance or suppress the channel and spatial information in the feature map to extract more discriminative features, thus improving the feature extraction capability.

In this block, CBAM is an attention mechanism module that combines the channel attention module (CAM) and the spatial attention module (SAM). CAM enhances the network's representation in the channel dimension by adaptivly learning the weights of each channel and fusing the important ones with weights. SAM utilizes the correlation between any two point features to mutually enhance the representation of their respective features, and, therefore, focuses more on spatial location features. CBAM first calculates the importance of each channel through CAM, then applies the channel attention weights to the feature map. Subsequently, the importance of each location is calculated by SAM to obtain the feature map, which captures the global dependency of features. The CBAM module is introduced to improve the model representation by focusing on the important parts of the input feature map, capturing the important features in the data, and improving the feature extraction capability and model performance.

The entire network structure consists of two convolutional layers, two CBAM modules, two max-

pooling layers, and two fully connected layers. Each convolutional layer is followed by a CBAM module, and a max-pooling layer is used after each CBAM, followed by two fully connected layers. The final length of the output sequences is kept as the same length of the outputs after 1D feature extraction.
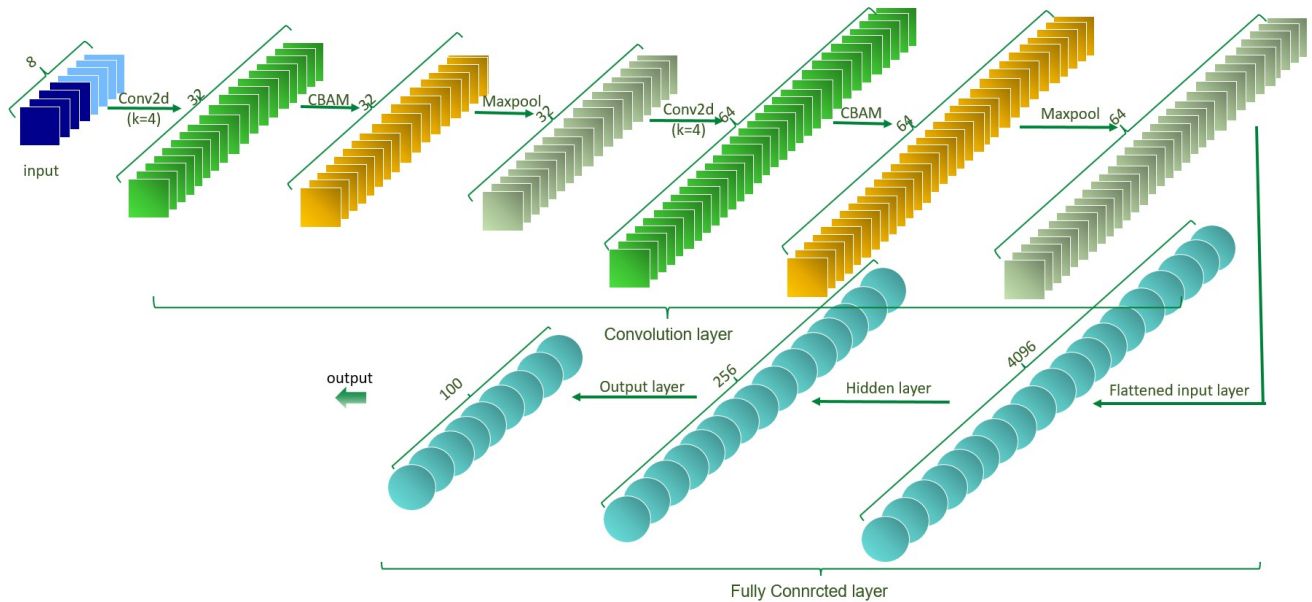
Figure 4. 2D feature extraction section.
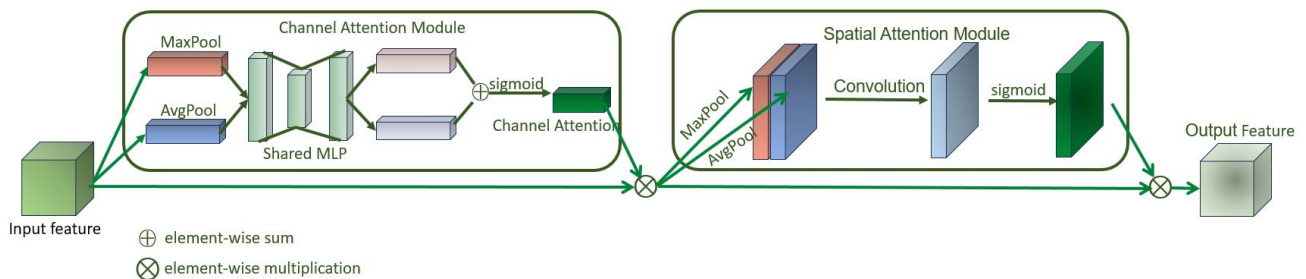
Figure 5. CBAM module.

3.3. Measuring sample similarity based on nonlinear mapping

Sample similarity measurement is the core issue of fault detection. As a typical few-shot method, siamese networks measure the similarity of two samples by calculating the $L1$ or $L2$ distance of the eigenvectors of them. However, because the bearing vibration signal has the characteristics of periodicity, multifrequency, nonlinearity and randomness, the direct $L1$ or $L2$ distance to the feature vector cannot effectively measure the similarity between samples. Metric learning is a solution for the above issue, which maps samples into the same embedding space and then calculates their similarity. Inspired by metric learning, this paper proposes a similarity measurement method mapping to the metric space and then calculates the $L1$ distance as shown in Eq (3.9).

Table 1. Parameters setting for WDCNN.

No.	Layer Type	Kernel Size/Stride	Kernel Number	Output Size (Width × Depth)
1	Convolution1	64 × 1/1 × 1	16	128 × 16
2	Pooling1	2 × 1/2 × 1	16	64 × 16
3	Convolution2	3 × 1/1 × 1	32	64 × 32
4	Pooling2	2 × 1/2 × 1	32	32 × 32
5	Convolution3	3 × 1/1 × 1	64	32 × 64
6	Pooling3	2 × 1/2 × 1	64	16 × 64
7	Convolution4	3 × 1/1 × 1	64	16 × 64
8	Pooling4	2 × 1/2 × 1	64	16 × 64
9	Convolution5	3 × 1/1 × 1	64	6 × 64
10	Pooling5	2 × 1/2 × 1	64	3 × 64
11	Fully-connected	100	1	100 × 1
12	Fully-connected	10	1	10 × 1

Table 2. Parameters setting for 2D-CNN.

No.	Layer Type	Kernel Size	Kernel Number	Output Size
1	Convolution1	7 × 7	64	44 × 44
2	Pooling1	2 × 2	64	21 × 21
3	Convolution2	5 × 5	32	18 × 18
4	Pooling2	2 × 2	32	8 × 8
5	Fully connected	10	1	10

$$f(x_i) = \frac{1}{1 + e^{-x_i}} \tag{3.8}$$

$$D\left(x_i, x_{i+1}\right) = \sum \left| f\left(x_i\right) - f\left(x_{i+1}\right)\right| \tag{3.9}$$

First of all, the feature vector output after feature extraction is mapped nonlinearly. We use the sigmoid activation function, which restricts the range of values of the feature vector between 0 and 1, normalizes the range of values of the features, and makes the features more comparable. Compared with the direct $L1$ distance to the feature vectors, the method of using the sigmoid activation function mapping and $L1$ distance calculation can more accurately measure the similarity between the samples, which improves the accuracy and stability of the similarity calculation.

The output is obtained by Eq (3.10), which represents the probability that the two input samples are the same:

$$P\left(x_i, x_{i+1}\right) = f\left(FC\left(D\left(x_i, x_{i+1}\right)\right)\right) \tag{3.10}$$

where FC is the fully connected layer.

4. Experiment

4.1. Experimental setup and data description

The code for the paper was implemented on a server equipped with two RTX4090Ti GPUs. The bearing failure dataset from Case Western Reserve University (CWRU) is used. The bearing failure dataset from CWRU contains different types of bearing failures in four states: normal operation, inner ring failure, outer ring failure, and rolling element failure. Each failure type simulates three single point failures of varying severity, with failure diameters of 0.07 inches, 0.14 inches, and 0.21 inches, for a total of 10 states [40]. The data contains the fan-side vibration data, the drive-side vibration data base vibration data, and the motor speed.

We compared the proposed method with SVM [10], WDCNN [40], 2DCNN [37], and a few-shot method [41]. The kernel function of SVM is a radial basis function that automatically adjusts the value of the kernel parameter according to the number of input features, with a penalty parameter of 1 and a "One-vs-One" decision function. The details of WDCNN [40] are shown in Table 1,and the details of the 2DCNN-based [37] approach are also given in Table 2. Both methods use a learning rate of 0.01, adopt cross-entropy loss as the loss function, and set epochs to 3000. The few-shot-based approach [41], which employs the WDCNN as the feature extraction portion of the twin network, uses the $L1$ distance to measure the similarity between samples. The learning rate of the method is set to 0.01, the loss function adopts the binary cross-entropy loss, and the epoch is set to 10,000. The learning rate and epoch of our method also refer to the few-shot-based approach, and we have carried out a number of experiments; we set the learning rate to 0.01, the loss function adopts the binary cross-entropy loss, and the epoch is set to 12,000.
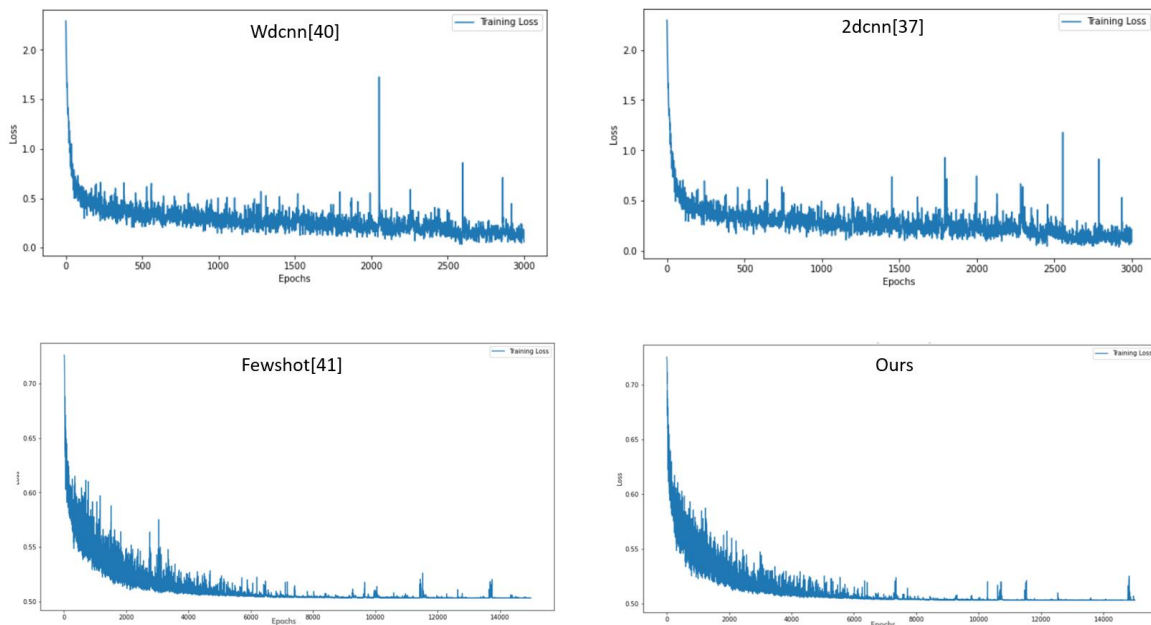

Figure 6. Training convergence curves.

4.2. Effect of number of training samples on performance

In the few-shot-based approach, the input is a sample pair coming from the same class or different classes, and the output is the probability of two input samples belonging to the same class. The test samples are then classified according to the most similar samples in the test set. Assuming that a test needs to be performed on x_t, the template set T contains samples from each class:

$$T = \{(X_1, Y_1) \ldots \ldots (X_i, Y_i)\} \tag{4.1}$$

$$M(x_t, (X_1, X_2, \ldots, X_i)) = argmax(P(x_t, x_m)), x_m \in T \tag{4.2}$$

$$y_t = y_m \tag{4.3}$$

where $p()$ denotes the computation of similarity between two samples, and y_m is the label corresponding to the sample with the highest similarity.

In the training phase, all experiments were conducted ten times to ensure the fairness of the experiments. Our method is compared with four classical methods, and to illustrate the performance of the method for bearing fault detection with different sample sizes, the sample size of each group is set to 60, 90, 120, 200, 300, 480, 720, 900, 1200, 1500. The training convergence curves of the sample size 200 are shown in Figure 6, and accuracy of all methods are given in Figure 7. Meanwhile, multi-category receiver operating characteristic curve (ROC) curves are generated for each category by calculating the true and false positive rates, then micro-averaging and macro-averaging are performed to obtain the micro-averaged area under the ROC curve (AUC) and macro-averaged AUC as an evaluation metric for the overall performance of the classifiers. The ROC curves are shown in Figure 8, and the results of training time, testing time, and F1-score are shown in Table 3.

Table 3. Time and F1-score.

	60-time-train	60-time-test	60-F1-score	200-time-train	200-time-test	200-F1-score
wdcnn [40]	324.10 s	0.676 s	0.491	117.70 s	1.214 s	0.612
cnn [37]	286.366 s	0.130 s	0.527	235.590 s	3.960 s	0.779
fewshot [41]	431.295 s	7.777 s	0.684	460.762 s	9.507 s	0.891
ours	602.958 s	9.992 s	0.716	1021.204 s	15.901 s	0.899
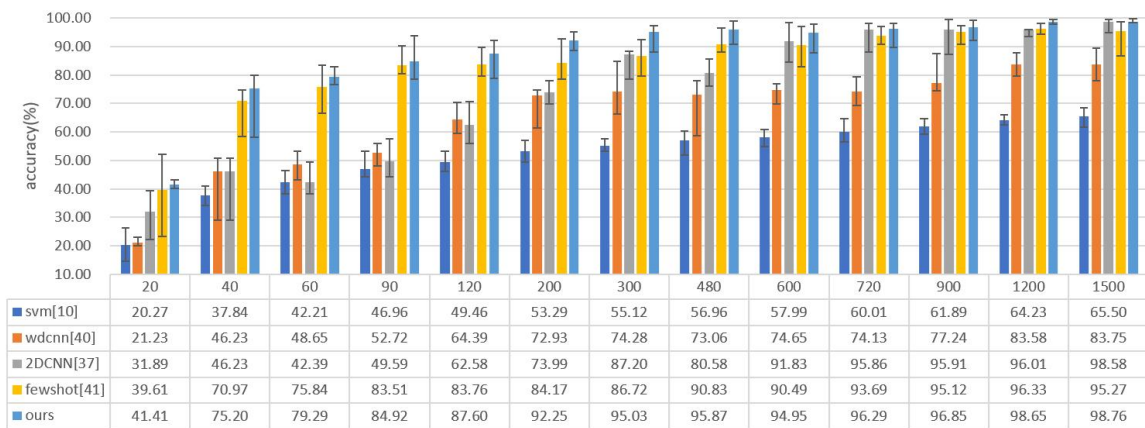
Figure 7. Comparison of experimental results.

	20	40	60	90	120	200	300	480	600	720	900	1200	1500
svm[10]	20.27	37.84	42.21	46.96	49.46	53.29	55.12	56.96	57.99	60.01	61.89	64.23	65.50
wdcnn[40]	21.23	46.23	48.65	52.72	64.39	72.93	74.28	73.06	74.65	74.13	77.24	83.58	83.75
2DCNN[37]	31.89	46.23	42.39	49.59	62.58	73.99	87.20	80.58	91.83	95.86	95.91	96.01	98.58
fewshot[41]	39.61	70.97	75.84	83.51	83.76	84.17	86.72	90.83	90.49	93.69	95.12	96.33	95.27
ours	41.41	75.20	79.29	84.92	87.60	92.25	95.03	95.87	94.95	96.29	96.85	98.65	98.76

Figure 8. ROC with 200 samples.

Our model performs better with higher average accuracy when the training sample size is 60, 90, 120, 200, 300, 480, 720, 900, 1200, 1500. As the number of training samples continues to increase, the accuracy of the algorithms will actually get closer. These performance comparisons show that our model effectively improves the accuracy of bearing fault detection in a few sample situations.

It can be seen that the latter two methods, which determine whether they belong to the same class or different classes by measuring the similarity of the input signals and then classifying them, are

much better than the other methods in terms of accuracy in the task under small sample conditions. Among them, our method works better than the other methods in the small sample condition of 20-1500, especially in the sample number of 300 with the baseline model than the improvement of 8.32%. This shows that our method is effective in the small sample condition and it can better capture the local features and implicit information in the small samples.

By looking at the ROC curve (in the classification task), it can be seen that our method has an ROC curve closer to the upper left corner compared to the other three methods. This trend suggests that our model is able to achieve a high true positive rate (TPR) while maintaining a low false positive rate (FPR) relative to the other methods, illustrating that our method is more accurate and effective in its ability to differentiate between different categories. In addition, the larger area under the ROC curve is also an important indicator for assessing the overall performance of the model, further emphasizing the superiority of our method. The analysis of the F1-score results shows that our method has a higher F1-score compared to the other three methods. This demonstrates that our method is more balanced and reliable in dealing with category imbalance or in scenarios with high requirements for both accuracy and coverage.

4.3. Comparative experiments in noisy environments

In this experiment, we discuss the performance in a noisy environment to simulate the variation of the operating conditions in the dataset. The signal-to-noise ratio is defined as in Equation 4.4. We train the model using the raw data provided by CWRU, with the number of samples set to 60, 90, 120, 200, 300, 480, 720, 900, 1200, 1500, and then test it with Gaussian white noise with an additive range of 2 dB to 10 dB.

$$SNR = 10log_{10}\left(\frac{P_s}{P_n}\right) \tag{4.4}$$

where P_s is the power of the signal and P_n is the power of the noise.

Table 4 shows the performance comparison of three different models in various noise environments. The results show that in most cases, our models are able to identify and utilize the valid information in the data more accurately than the other models, especially when confronted with noise, and our models achieve better test scores. The negative impact of noise on model performance is highlighted by the fact that the performance of all models decreases as the noise level increases. However, our model shows a clear advantage in this challenge, especially when the sample size is increased, by being able to distinguish between signal and noise more efficiently, which improves the robustness of the model. In addition, the experimental data also shows that even in a small sample size and noisy environment, our method maintains good performance, exhibiting greater robustness and generalization ability than the other two methods.

4.4. Comparison experiment under new working conditions

In this experiment, we evaluated the test results of sample size for each training model at a sample size of 60, where the training set and the test set are different rotational speeds as shown in Table 5. Two of the working conditions are used for training, and the new working conditions are tested to evaluate the effectiveness of the detection of the emergence of the new conditions under the few sample conditions. In this experiment, we aim to evaluate the detection effectiveness of different training models in the face of new working conditions when dealing with less sample datasets.
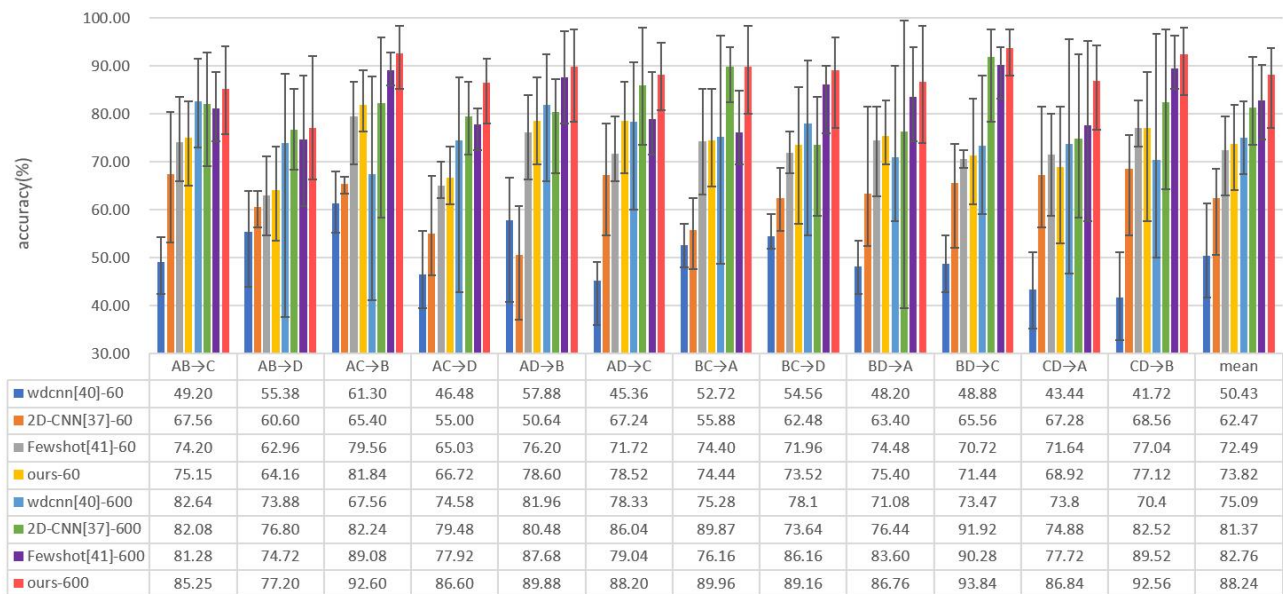
Table 4. Test results under noise conditions.

model	num	SNR (dB)				
		2	4	6	8	10
fewshot[41]	20	34.91	36.56	37.72	37.97	39.08
	40	61.49	65.97	68.30	69.44	70.39
	60	62.59	68.25	70.87	73.21	74.13
	90	71.19	76.79	80.13	81.57	82.44
	120	70.96	76.81	80.57	82.01	82.45
	200	67.02	77.09	80.12	82.84	83.93
	300	61.19	71.83	78.47	83.20	85.23
	480	70.61	80.35	85.05	88.19	89.92
	600	64.85	77.19	83.95	88.24	89.55
	720	63.31	77.61	85.45	89.91	91.89
	900	58.87	74.91	85.13	90.81	93.05
	1200	61.45	78.52	88.16	92.73	94.77
	1500	53.00	71.75	85.03	90.67	93.69
wdcnn[40]	20	20.98	21.12	21.22	21.17	21.12
	40	32.38	37.29	41.62	44.30	44.84
	60	44.57	46.60	47.05	47.61	48.40
	90	45.31	48.44	50.48	51.23	51.71
	120	44.41	46.59	48.03	48.49	49.21
	200	62.11	67.00	69.64	71.36	72.16
	300	66.11	69.01	72.44	73.45	74.03
	480	63.88	68.03	70.10	71.50	72.40
	600	61.43	67.81	71.39	73.21	73.92
	720	63.89	69.39	71.73	72.99	73.49
	900	71.11	74.65	75.63	76.70	76.79
	1200	64.73	74.00	78.36	81.00	82.27
	1500	67.01	75.58	79.85	81.96	82.55
ours	20	39.00	39.92	40.22	40.00	41.02
	40	63.58	70.16	72.49	73.68	74.50
	60	62.76	73.36	76.85	78.17	78.74
	90	69.96	76.61	80.72	83.32	84.23
	120	63.33	73.48	80.68	84.20	85.88
	200	64.64	77.16	85.8	89.77	91.28
	300	64.99	79.41	87.49	91.89	93.52
	480	71.21	84.04	91.28	94.33	95.21
	600	61.45	79.47	88.61	92.37	94.01
	720	68.56	84.20	91.11	94.11	95.65
	900	65.87	83.05	91.01	94.96	96.41
	1200	68.60	82.20	92.09	96.23	97.95
	1500	67.21	83.59	91.57	95.94	97.23

Table 5. Dataset labels and their corresponding speeds.

Dataset	A	B	C	D
Speed	1730	1750	1772	1797

According to Figure 9, our method shows high average accuracy in all 12 different test cases. In particular, when the training sample size is 60, our method outperforms the best results of other methods by an average accuracy of 1.33%. This result emphasizes the effectiveness of our method in learning scenarios with few samples. With only 60 samples, the models are able to effectively learn and adapt to new working conditions, which demonstrates the learning efficiency and generalization ability of our method. The performance of all models improves as the sample size increases, but our method shows more significant performance gains. When the sample size is increased to 600, at this point, the average accuracy of our method is 5.48% higher than the best results of the other methods. These results indicate that our method has better performance when dealing with less sample datasets, especially when faced with new working conditions. The experimental results in this section show that our method has significant advantages in dealing with less sample datasets and in adapting to new working conditions.

	AB→C	AB→D	AC→B	AC→D	AD→B	AD→C	BC→A	BC→D	BD→A	BD→C	CD→A	CD→B	mean
wdcnn[40]-60	49.20	55.38	61.30	46.48	57.88	45.36	52.72	54.56	48.20	48.88	43.44	41.72	50.43
2D-CNN[37]-60	67.56	60.60	65.40	55.00	50.64	67.24	55.88	62.48	63.40	65.56	67.28	68.56	62.47
Fewshot[41]-60	74.20	62.96	79.56	65.03	76.20	71.72	74.40	71.96	74.48	70.72	71.64	77.04	72.49
ours-60	75.15	64.16	81.84	66.72	78.60	78.52	74.44	73.52	75.40	71.44	68.92	77.12	73.82
wdcnn[40]-600	82.64	73.88	67.56	74.58	81.96	78.33	75.28	78.1	71.08	73.47	73.8	70.4	75.09
2D-CNN[37]-600	82.08	76.80	82.24	79.48	80.48	86.04	89.87	73.64	76.44	91.92	74.88	82.52	81.37
Fewshot[41]-600	81.28	74.72	89.08	77.92	87.68	79.04	76.16	86.16	83.60	90.28	77.72	89.52	82.76
ours-600	85.25	77.20	92.60	86.60	89.88	88.20	89.96	89.16	86.76	93.84	86.84	92.56	88.24

Figure 9. Comparative experiment under new working conditions.

4.5. Ablation analysis of the proposed method

In Table 6, we show the results of model ablation for sample sizes of 60 and 600 to verify the contribution of each module.

(1) Item 1 demonstrates the effectiveness of the preprocessing block. We observe a decrease in accuracy of 0.49% and 4.7% in the noiseless condition compared to the full model, which suggests that by adding channels through our preprocessing method, more input information can be maintained in the convolution operation. This reduces information loss and provides a more comprehensive repre-

sentation of features, which facilitates bearing fault detection.

(2) Items 2 and 3 demonstrate the partial effectiveness of feature extraction block. We observe that the accuracy without 1DCNN decreases by 0.43% and 2.62% compared to the full model under noiseless condition, and the accuracy without 1DCNN decreases by 2.53% and 3.11% compared to the full model, which shows that our approach of using a mixture of one-dimensional convolution and two-dimensional convolution, and the attentional mechanism is effective. We demonstrate the advantage of the multidimensional feature extraction method over single dimension.

(3) Item 4 demonstrates the results of using the $L1$ method to measure the similarity between samples decreased by 5.3% and 3.11% compared to the original in the noiseless condition. The method of using nonlinear mapping is more effective than directly calculating the $L1$ distance, which demonstrates that our method has a greater advantage over the measurement of similarity between samples.

Table 6. Ablation results of the proposed method.

Num	method	exp	SNR(dB)					
			2	4	6	8	10	None
0	Original model	60	62.76	73.36	76.85	78.17	78.74	79.29
		600	61.45	79.47	88.61	92.37	94.01	94.95
1	No -preprocess	60	60.33	68.91	73.57	76.77	77.00	78.80
		600	60.27	75.69	83.64	87.03	88.59	90.25
2	No-1D-CNN	60	47.40	66.00	75.60	77.89	77.79	78.86
		600	46.45	67.68	82.26	88.41	90.02	92.33
3	No-2D-CNN+CBAM	60	62.36	68.67	70.87	73.24	75.21	76.74
		600	54.65	78.35	83.95	88.24	90.34	91.84
4	No-Nonlinear mapping	60	49.13	61.17	69.27	72.11	73.51	73.99
		600	42.57	61.77	78.47	89.21	92.56	94.49

5. Conclusions

The detection and diagnosis of rolling bearing fault is very important for the safe operation of rotating machinery. Bearing fault may cause great economic losses and even endanger the safety of personnel, so it is very important to find and diagnose it in time. However, it is difficult to collect enough bearing fault data under all working conditions. In order to solve this problem, this paper proposes a bearing fault diagnosis method based on the multidimensional convolution and attention mechanism, which uses median filtering, mean filtering, and convolution operation to preprocess the original data, and uses 1D-CNN and 2DCNN+CBAM to realize efficient feature extraction. At the same time, the similarity between samples is measured by nonlinear mapping of feature vectors into metric space. The experimental results verify the effectiveness of this method and have broad industrial application prospects.

In future work, we will focus on developing more lightweight and efficient models for deployment on mobile terminals and edge devices, making the models lighter and more efficient to handle the complexity and diversity of faults that may occur in real industrial scenarios. In addition, in the network

feature extraction part, consideration can be given to structures such as lightweight network structures or deep separable convolutions to reduce the complexity and computational overhead of the model while maintaining the ability to efficiently extract bearing fault features. Furthermore, exploring other network structures suitable for small sample learning that can diagnose unknown types of bearing faults occurring in real industrial scenarios could be beneficial.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

Conflict of interest

The authors declare there is no conflict of interest.

References

1. N. W. Nirwan, H. B. Ramani, Condition monitoring and fault detection in roller bearing used in rolling mill by acoustic emission and vibration analysis, *Mater. Today Proc.*, **51** (2022), 344–354. https://doi.org/10.1016/j.matpr.2021.05.447

2. S. Rajabi, M. S. Azari, S. Santini, F. Flammini, Fault diagnosis in industrial rotating equipment based on permutation entropy, signal processing and multi-output neuro-fuzzy classifier, *Expert Syst. Appl.*, **206** (2022), 117754. https://doi.org/10.1016/j.eswa.2022.117754

3. E. A. Burda, G. V. Zusman, I. S. Kudryavtseva, A. P. Naumenko, An overview of vibration analysis techniques for the fault diagnostics of rolling bearings in machinery, *Shock Vib.*, **2022** (2022). https://doi.org/10.1155/2022/6136231

4. J. Gu, Y. Peng, H. Lu, X. Chang, G. Chen, A novel fault diagnosis method of rotating machinery via VMD, CWT and improved CNN, *Measurement*, **200** (2022), 111635. https://doi.org/10.1016/j.measurement.2022.111635

5. J. Pacheco-Chérrez, J. A. Fortoul-Díaz, F. Cortés-Santacruz, L. M. Aloso-Valerdi, D. I. Ibarra-Zarate, Bearing fault detection with vibration and acoustic signals: Comparison among different machine leaning classification methods, *Eng. Fail. Anal.*, **139** (2022), 106515. https://doi.org/10.1016/j.engfailanal.2022.106515

6. K. Berg-Sørensen, H. Flyvbjerg, Power spectrum analysis for optical tweezers, *Rev. Sci. Instrum.*, **75** (2004), 594–612. https://doi.org/10.1063/1.1645654

7. R. B. Randall, A history of cepstrum analysis and its application to mechanical problems, *Mech. Syst. Signal Process.*, **97** (2017), 3–19. https://doi.org/10.1016/j.ymssp.2016.12.026

8. A. R. Al-Obaidi, H. J. Towsyfyan, An experimental study on vibration signatures for detecting incipient cavitation in centrifugal pumps based on envelope spectrum analysis, *J. Appl. Fluid Mech.*, **12** (2019), 2057–2067. https://doi.org/10.29252/JAFM.12.06.29901

9. N. Peifeng, Z. Jun, Z. Gang, Study on application of wavelet transform technique to turbine generator fault diagnosis, *Chin. J. Sci. Instrum.*, **28** (2007), 188. https://doi.org/10.19650/j.cnki.cjsi.2007.01.039

10. P. Konar, P. Chattopadhyay, Bearing fault detection of induction motor using wavelet and support vector machines (SVMs), *Appl. Soft Comput.*, **11** (2011), 4203–4211. https://doi.org/10.1016/j.asoc.2011.03.014

11. N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, et al., The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis, *Proc. R. Soc. London*, **454** (1998), 903–995. https://doi.org/10.1098/rspa.1998.0193

12. C. Song, S. Liu, G. Han, P. Zeng, H. Yu, Q. Zheng, Edge-intelligence-based condition monitoring of beam pumping units under heavy noise in industrial internet of things for industry 4.0, *IEEE Int. Things J.*, **10** (2023), 3037–3046. https://doi.org/10.1109/JIOT.2022.3141382

13. S. Liu, C. Song, T. Wu, P. Zeng, A lightweight fault diagnosis method of beam pumping units based on dynamic warping matching and parallel deep network, *IEEE Trans. Syst., Man, Cybern.*, **54** (2023), 1622–1632. https://doi.org/10.1109/TSMC.2023.3328731

14. Q. Cui, Z. Li, J. Yang, B. Liang, Rolling bearing fault prognosis using recurrent neural network, in *2017 29th Chinese Control And Decision Conference (CCDC)*, (2017), 1196–1201. https://doi.org/10.1109/CCDC.2017.7978700

15. L. Yu, J. Qu, F. Gao, Y. Tian, A novel hierarchical algorithm for bearing fault diagnosis based on stacked LSTM, *Shock Vib.*, **2019** (2019). https://doi.org/10.1155/2019/2756284

16. C. Song, P. Zeng, Z. Wang, T. Li, L. Qiao, L. Shen, Image forgery detection based on motion blur estimated using convolutional neural network, *IEEE Sens. J.*, **19** (2019), 11601–11611. https://doi.org/10.1109/JSEN.2019.2928480

17. M. Ye, X. Yan, D. Jiang, L. Xiang, N. Chen, MIFDELN: A multi-sensor information fusion deep ensemble learning network for diagnosing bearing faults in noisy scenarios, *Knowl.-Based Syst.*, **284** (2024), 111294. https://doi.org/10.1016/j.knosys.2023.111294

18. X. Yan, W. J. Yan, Y. Xu, K. V. Yuen, Machinery multi-sensor fault diagnosis based on adaptive multivariate feature mode decomposition and multi-attention fusion residual convolutional neural network, *Mech. Syst. Signal Process.*, **202**, (2023), 110664.https://doi.org/10.1016/j.ymssp.2023.11066

19. X. Yan, D. She, Y. Xu, Deep order-wavelet convolutional variational autoencoder for fault identification of rolling bearing under fluctuating speed conditions, *Expert Syst. Appl.*, **216** (2023), 119479. https://doi.org/10.1016/j.eswa.2022.119479

20. H. Liu, H. Zhao, J. Wang, S. Yuan, W. Feng, LSTM-GAN-AE: A promising approach for fault diagnosis in machine health monitoring, *IEEE Trans. Instrum. Meas.*, **71** (2021), 1–13. https://doi.org/10.1109/TIM.2021.3135328

21. J. Yang, J. Liu, J. Xie, C. Wang, T. Ding, Conditional GAN and 2-D CNN for bearing fault diagnosis with small samples, *IEEE Trans. Instrum. Meas.*, **70** (2021), 1–12. https://doi.org/10.1109/TIM.2021.3119135

22. G. Yang, C. Song, Z. Yang, S. Cui, Bubble detection in photoresist with small samples based on GAN augmentations and modified YOLO, *Eng. Appl. Artif. Intell.*, **123** (2023), 106224. https://doi.org/10.1016/j.engappai.2023.106224

23. C. Song, W. Xu, Z. Wang, S. Yu, Z. Ju, Analysis on the impact of data augmentation on target recognition for UAV-based transmission line inspection, *Complexity*, **2020** (2020). https://doi.org/10.1155/2020/3107450

24. O. Bohdal, Y. Tian, Y. Zong, R. Chavhan, D. Li, H. Gouk, et al., Meta omnium: A benchmark for general-purpose learning-to-learn, in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2023), 7693–7703. https://doi.org/10.1109/CVPR52729.2023.00743

25. K. Song, J. Han, G. Cheng, J. Lu, F. Nie, Adaptive neighborhood metric learning, *IEEE Trans. Pattern Anal. Mach. Intell.*, **44** (2022), 4591-4604. https://doi.org/10.1109/TPAMI.2021.3073587

26. Z. Wu, H. Jiang, K. Zhao, X. Li, An adaptive deep transfer learning method for bearing fault diagnosis, *Measurement*, **151** (2020), 107227. https://doi.org/10.1016/j.measurement.2019.107227

27. J. Zhu, N. Chen, C. Shen, A new deep transfer learning method for bearing fault diagnosis under different working conditions, *IEEE Sens. J.*, **20** (2019), 8394–8402. https://doi.org/10.1109/jsen.2019.2936932

28. B. Yang, Y. Lei, F. Jia, S. Xing, An intelligent fault diagnosis approach based on transfer learning from laboratory bearings to locomotive bearings, *Mech. Syst. Signal Process.*, **122** (2019), 692–706. https://doi.org/10.1016/j.ymssp.2018.12.051

29. X. Li, Y. Hu, M. Li, J. Zheng, Fault diagnostics between different type of components: A transfer learning approach, *Appl. Soft Comput.*, **86**, (2020), 105950. https://doi.org/10.1016/j.asoc.2019.105950 Get rights and content

30. Y. F. Li, M. Zuo, K. Feng, Y. J. Chen, Detection of bearing faults using a novel adaptive morphological update lifting wavelet, *Chin. J. Mech. Eng.*, **30** (2017), 1305–1313. https://doi.org/10.1007/s10033-017-0186-1

31. Q. Fu, B. Jing, P. He, S. Si, Y. Wang, Fault feature selection and diagnosis of rolling bearings based on EEMD and optimized Elman_AdaBoost algorithm, *IEEE Sens. J.*, **18** (2018), 5024–5034. https://doi.org/10.1109/JSEN.2018.2830109

32. J. Zheng, S. Huang, H. Pan, J. Tong, C. Wang, Q. Liu, Adaptive power spectrum Fourier decomposition method with application in fault diagnosis for rolling bearing, *Measurement*, **183** (2021), 109837. https://doi.org/10.1016/j.measurement.2021.109837

33. K. Feng, Q. Ni, M. Beer, H. Du, C. Li, A novel similarity-based status characterization methodology for gear surface wear propagation monitoring, *Tribol. Int.*, **174** (2022), 107765.

34. K. Feng, J. C. Ji, Y. Zhang, Q. Ni, Z. Liu, M. Beer, Digital twin-driven intelligent assessment of gear surface degradation, *Mech. Syst. Signal Process.*, **186** (2023), 109896. https://doi.org/10.1016/j.ymssp.2022.109896

35. Q. Ni, J. C. Ji, B. Halkon, K. Feng, A. K. Nandi, Physics-informed pesidual network (PIResNet) for rolling element bearing fault diagnostics, *Mech. Syst. Signal Process.*, **200** (2023), 110544. https://doi.org/10.1016/j.ymssp.2023.110544

36. D. Peng, Z. Liu, H. Wang, Y. Qin, L. Jia, A novel deeper one-dimensional CNN with residual learning for fault diagnosis of wheelset bearings in high-speed trains, *IEEE Access*, **7** (2018), 10278–10293. https://doi.org/10.1109/ACCESS.2018.2888842

37. X. Peng, B. Zhang, D. Gao, Research on fault diagnosis method of rolling bearing based on 2DCNN, in *2020 Chinese Control And Decision Conference (CCDC)*, (2020), 693–697. https://doi.org/10.26914/c.cnkihy.2020.033919

38. D. Wang, Q. Guo, Y. Song, S. Gao, Y. Li, Application of multiscale learning neural network based on CNN in bearing fault diagnosis, *J. Signal Process. Syst.*, **91** (2019), 1205–1217. https://doi.org/10.1007/s11265-019-01461-w

39. A. Khorram, M. Khalooei, M. Rezghi, End-to-end CNN+ LSTM deep learning approach for bearing fault diagnosis, *Appl. Intell.*, **51** (2021), 736–751. https://doi.org/10.1007/s10489-020-01859-1

40. W. Zhang, G. Peng, C. Li, Y. Chen, Z. Zhang, A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals, *Sensors*, **17**, (2017), 425. https://doi.org/10.3390/s17020425

41. A. Zhang, S. Li, Y. Cui, W. Yang, R. Dong, J. Hu, Limited data rolling bearing fault diagnosis with few-shot learning, *IEEE Access*, **7** (2019), 110895–110904. https://doi.org/10.1109/ACCESS.2019.2934233