Article

Application of Deep Learning in Fault Diagnosis of Rotating Machinery

Wanlu Jiang ^{1,2}, Chenyang Wang ^{1,2,*}, Jiayun Zou ^{1,2} and Shuqing Zhang ³

¹ Hebei Provincial Key Laboratory of Heavy Machinery Fluid Power Transmission and Control, Yanshan University, Qinhuangdao 066004, China; wljiang@ysu.edu.cn (W.J.); zjy384733369@163.com (J.Z.)

² Key Laboratory of Advanced Forging & Stamping Technology and Science, Yanshan University, Ministry of Education of China, Qinhuangdao 066004, China

³ School of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China; zhshq-yd@163.com

* Correspondence: wangcy@stumail.ysu.edu.cn

Abstract: The field of mechanical fault diagnosis has entered the era of “big data”. However, existing diagnostic algorithms, relying on artificial feature extraction and expert knowledge are of poor extraction ability and lack self-adaptability in the mass data. In the fault diagnosis of rotating machinery, due to the accidental occurrence of equipment faults, the proportion of fault samples is small, the samples are imbalanced, and available data are scarce, which leads to the low accuracy rate of the intelligent diagnosis model trained to identify the equipment state. To solve the above problems, an end-to-end diagnosis model is first proposed, which is an intelligent fault diagnosis method based on one-dimensional convolutional neural network (1D-CNN). That is to say, the original vibration signal is directly input into the model for identification. After that, through combining the convolutional neural network with the generative adversarial networks, a data expansion method based on the one-dimensional deep convolutional generative adversarial networks (1D-DCGAN) is constructed to generate small sample size fault samples and construct the balanced data set. Meanwhile, in order to solve the problem that the network is difficult to optimize, gradient penalty and Wasserstein distance are introduced. Through the test of bearing database and hydraulic pump, it shows that the one-dimensional convolution operation has strong feature extraction ability for vibration signals. The proposed method is very accurate for fault diagnosis of the two kinds of equipment, and high-quality expansion of the original data can be achieved.

Keywords: fault diagnosis; 1D-CNN; 1D-DCGAN; bearing; hydraulic pump; small sample size

Citation: Jiang, W.; Wang, C.; Zou, J.; Zhang, S. Application of Deep Learning in Fault Diagnosis of Rotating Machinery. *Processes* **2021**, *9*, 919. <https://doi.org/10.3390/pr9060919>

Academic Editors: Jie Zhang and Simant Upreti

Received: 29 March 2021

Accepted: 20 May 2021

Published: 24 May 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Rotating machinery is the most diffusely used mechanical equipment in industrial production. Bearing, as a common part of rotating machinery equipment, plays an important role in machinery, power system and other large industrial equipment. Similarly, the hydraulic pump is also a common and essential rotating mechanical component. Hydraulic pump failure will cause the entire hydraulic system to not work properly. With the development of rotating machinery equipment towards the direction of high-grade, precision and advanced properties, it must rely on the theory and method of fault diagnosis to escort items, which raises higher requirements for fault diagnosis in the era of industrial big data [1,2].

In recent years, mechanical fault diagnosis technology has been rapidly developed. Researchers and engineering experts have actively explored fault mechanism and symptom connection, signal processing and feature extraction, recognition and classification, and intelligent decision-making, and proposed a large number of methods and technology. In terms of fault mechanism, the fault mechanism of rotating machinery has been effectively explored based on dynamic behavior [3,4]. In the traditional signal processing and feature extraction technology, fault diagnosis methods are mainly divided into three aspects:

time domain analysis [5], frequency domain analysis [6], and time-frequency domain analysis [7–9]. With the development of computer science, intelligent methods such as artificial intelligence, pattern recognition, and machine learning have been continuously applied to mechanical fault diagnosis tasks [10].

In the past few years, deep learning in academia and industry develops rapidly. By simulating the brain learning process to build a deep level model, combined with huge amounts of training data, to study the implicit characteristics of the data, the recognition accuracy rate in many traditional recognition tasks is significantly improved and its superb ability is demonstrated in dealing with a large amount of data, and recognition of complex task [11–13].

The convolutional neural network (CNN), as an important branch of deep learning, is mainly applied to the feature extraction of 2D and 3D image sequences [14,15]. Many scholars have introduced deep learning into the fault diagnosis of rotating machinery equipment [16–20]. Some studies combined other algorithms with CNN, and CNN was used as a classifier or feature extractor [21–23]. CNN's powerful feature self-extraction capabilities are not used in end-to-end fault diagnosis. In recent years, some scholars have taken vibration signals as research objects, introduced CNN into fault diagnosis of bearings and hydraulic pumps, and achieved good results by converting vibration signals into two-dimensional time-frequency diagrams for fault diagnosis [24–29]. With vibration signals as a one-dimensional time-series signal, the data points at per moment are correlated. If the vibration signal is directly converted into a two-dimensional form, the spatial correlation in the original sequence will be broken, which may cause the loss of fault information. At present, most CNN-based fault diagnosis methods do not directly obtain data information from the original signals, and the powerful characteristic self-learning ability of CNN is not fully utilized, which limits the improvement of fault identification rate.

In recent years, there have been numerous studies on intelligent diagnosis of mechanical faults. However, these studies are generally based on the assumption that there is sufficient available monitoring data and requires mechanical monitoring data for intelligent diagnosis model training: training data samples are balanced, typical fault information is abundant, and category labeling information is sufficient. In practical engineering, these assumptions are difficult to satisfy. The number of normal samples is much higher than that of fault samples due to the occasional failure of rotating machinery equipment. The fault diagnosis model trained by imbalanced data samples has poor generalization ability, which is bound to cause wrong judgment to the real fault data. The judgment of fault signals as normal signals may cause enormous economic losses. Some scholars have improved the algorithm itself to improve the accuracy rate of rotating machinery fault identification based on imbalanced samples [30–33]. In the classification and identification tasks based on deep learning, the two problems that have the greatest impact on the accuracy rate are the quality of data and the performance of the algorithm. Expansion of small sample data is a more effective and direct way to deal with the identification task of imbalanced samples. The key to the problem is whether small sample data of high quality can be generated. To solve this problem, a diagnostic model that can handle imbalanced data and expand a small sample size samples is urgently required. In deep learning, another model—the generative adversarial network (GAN), may be a very effective way to deal with the imbalanced data set [34,35]. Some researchers used GAN to generate two-dimensional diagrams of original vibration signals to expand the data set and improve the identification accuracy rate [36,37]. However, it does not expand directly from the original data, and does not fully mine the feature information in the original one-dimensional time-series data. At the same time, the data form of the image also limits the application range of the generated data.

In the field of fault diagnosis, in the era of big data, there are two major problems: the difficulty of extracting features from massive data and the imbalance of samples. This paper will build an end-to-end diagnosis model, and puts forward an intelligent fault diagnosis method based on a one-dimensional convolutional neural network, to take full advantage of the depth network of CNN to achieve self-learning features, and can automatically

complete signal feature extraction and fault identification, with original vibration data is used as the input of the model, and the output of the model as the diagnostic results. Then, a sample expansion method is proposed, which integrates the one-dimensional convolutional neural network into the GAN model, constructs the one-dimensional deep convolutional generative adversarial network to generate the original vibration data, and solves the problem of imbalanced sample. Finally, through the bearing database and hydraulic pump test, the verification and analysis are launched.

2. 1D-CNN Intelligent Fault Diagnosis Method

Convolutional neural networks have been applied in fault diagnosis, but most of them only extend from image identification to fault feature map identification, or only use CNN as classifier in the last step of the fault diagnosis. However, the nonlinear features contained in the original signals are not extracted by CNN, even though it has a robust feature extraction ability. Therefore, this paper constructs a fault diagnosis method based on one-dimensional convolutional neural network (1D-CNN).

2.1. Convolutional Neural Network

CNN is a typical feed-forward neural network. A typical CNN usually includes input layer, convolutional layer, pooling layer, fully connected layer and output layer.

2.1.1. Convolutional Layer

The convolution operation improves traditional neural networks through three crucial ideas: parameter sharing, equivariant representations and sparse interactions [38]. The convolution kernel performs a convolution operation on the feature vector output by the previous layer and uses a nonlinear activation function to construct the output feature vector. The output of each layer is the convolution result of multiple input features. Its mathematical model can be described as:

$$x_j^l = f \left(\sum_{i \in M_j} x_i^{l-1} \times k_{ij}^l + b_j^l \right) \quad (1)$$

where M_j is the input eigenvector, l is the l -th layer in the network, k is the convolution kernel, b is the network bias, x_j^l is the output of the l -th layer, and x_i^{l-1} is the input of the l -th layer.

2.1.2. Pooling Layer

Pooling is a form of nonlinear down-sampling, which reduces the amount of calculation by reducing network parameters and can control overfitting to a certain extent. Currently, a pooling layer is added after the convolutional layer. Maximum pooling is to divide the input layer into different regions with non-overlapping rectangular boxes [39]. The maximum number of rectangular boxes is taken as the output layer. The transformation function of maximum pooling is expressed as:

$$P_i^{l+1}(j) = \max_{(j-1)V+1 \leq n \leq jV} \{q_i^l(n)\} \quad (2)$$

where $q_i^l(n)$ represents the value of the n -th neuron in the i -th eigenvector of the l -th layer, $n \in [(j-1)V+1, jV]$, V is the width of the pooling area, $P_i^{l+1}(j)$ represents the corresponding value of neurons in the $(l+1)$ -th layer.

2.1.3. Fully Connected Layer

The fully connected layer is a traditional feed-forward neural network. After that, the Softmax function is used as the activation function at the output to solve the multi-classification problem [40]. The fully connected layer plays the role of mapping the learned

“distributed feature representation” to the sample label space. The specific expression is as follows

$$\mathbf{O} = f(\mathbf{w}_o \mathbf{f}_v + \mathbf{b}_o) \quad (3)$$

where $\mathbf{f}_v$ is the eigenvector; $\mathbf{w}_o$ is weight matrix and $\mathbf{b}_o$ is bias vector.

2.2. Establishment of 1D-CNN Intelligent Fault Diagnosis Method

The structure of the one-dimensional convolutional neural network (1D-CNN) constructed in this paper is shown in Figure 1. It includes three parts: input layer, feature extraction layer, and classification layer. The input layer is the direct input after segmenting the original data. The feature extraction layer includes three convolutional layers and three pooling layers. It receives data from the input layer and extracts the features of the original vibration signal. The pooling layer selected the maximum pooling operator to reduce the dimension of the feature vector and improve the robustness of nonlinear features. The classification layer is composed of two fully connected layers. The number of neurons in the second fully connected layer is the same as the number of fault labels. The Softmax regression classifier is used to achieve classification of output.

Figure 1. The 1D-CNN structure diagram.

The loss function of the model is to evaluate the difference between the Softmax output probability distribution obtained by training and the true distribution. The training goal is to minimize the loss function. This article chooses the cross-entropy loss function. The formula is as follows:

$$E = -\frac{1}{n} \sum_{i=1}^n [y_i \ln a_i + (1 - y_i) \ln(1 - a_i)] \quad (4)$$

where n is the number of samples, a is the predicted value, and y is the true value.

RMSProp algorithm combined with Nesterov momentum is used to minimize the loss function during the training. Empirically, RMSProp is an effective and practical deep neural network optimization algorithm.

2.3. Experimental Verification

In order to verify the effectiveness of the 1D-CNN diagnosis method proposed in this paper, the deep groove ball bearing vibration data set of the Open Bearing Data Set, from CWRU in the United States, is used. The bearing failure simulation test bench is shown in

Figure 2. Using EDM technology, single-point failures have been arranged on the inner ring, outer ring, and rolling body of the bearing. The fault diameters are 7 mils, 14 mils, 21 mils, 28 mils, and 40 mils.

Figure 2. Bearing fault simulation test bench.

The acceleration data at the drive end was used as the experimental data. Vibration signals sampled at a sampling frequency of 12 kHz and at a load of 2HP were selected, including four states: normal state, inner ring fault, outer ring fault and rolling body fault. Three different fault degrees were then selected for each fault type, to be used as fault samples. Taking into consideration that the deep learning training model needs a large amount of data to support it, a sample expansion of the other nine fault data, except for the normal data, was carried out. The expansion mode is shown in Figure 3. Each sample selected 1024 sampling points from the original vibration signal, and each sample maintains a 50% overlapped with the previous sample. The sample contained ten types of bearing states. The composition of experimental sample is shown in Table 1. Each bearing state was randomly selected to form a training set, verification set and testing set, with a ratio of 3:1:1.

Figure 3. Vibration signal expansion mode.

Table 1. Composition of experimental samples.

Sample Type	Sample Length	Number of Samples	Label
Normal state	1024	400	0
Inner ring fault (7 mils)	1024	400	1
Outer ring fault (7 mils)	1024	400	2
Rolling body fault (7 mils)	1024	400	3
Inner ring fault (14 mils)	1024	400	4
Outer ring fault (14 mils)	1024	400	5
Rolling body fault (14 mils)	1024	400	6
Inner ring fault (21 mils)	1024	400	7
Outer ring fault (21 mils)	1024	400	8
Rolling body fault (21 mils)	1024	400	9

2.3.1. 1D-CNN Parameter Selection

A single GPU, model number is RTX2080ti, was used for the training of the 1D-CNN model constructed in this article. In terms of the choice of the number of convolutional layers, theoretically, the depth determines the expressive ability of the network, and the deeper the network, the stronger the learning ability. However, the optimization problems, activation function problems, and gradient problems brought about by more layers will become more and more complicated. On the issue of convolution kernel size, the convolutional kernel of different sizes will affect the size of the field of view. After the experiment was analyzed, it was found that in the CNN, number of convolutional layers and convolution kernel size are the key factors that determine the performance of the network. This part only discusses the influence of the number of convolutional layers and convolution kernel size on the model. The goal is to explore a more compact and efficient model structure (rather than a deep network) that is better suited to real-time and big data fault diagnosis. Before determining the final network structure, construct five network structures, and the structure of each model is consistent except for the number of convolutional layers. The number of neurons in the penultimate fully connected layer is 256. For each structure, a maximum pooling layer is added after the first convolutional layer and the last convolutional layer. All other hyperparameters are adjusted to be optimal. The five network structures, training time, and testing set identification accuracy rate are shown in Table 2.

Table 2. Performance comparison of five network structures.

	Structure 1	Structure 2	Structure 3	Structure 4	Structure 5
Number of convolutional layers	3	3	9	9	15
Number of convolutional kernels	(256, 512, 1024)	(256, 512, 1024)	(64, 128, 128, 128, 128, 256, 256, 512, 1024)	(64, 128, 128, 128, 128, 256, 256, 512, 1024)	(32, 32, 32, 64, 64 64 128 128, 128, 256, 256, 256, 512, 512, 1024)
Convolutional kernel size	(7 × 1, 5 × 1, 5 × 1)	(3 × 1, 3 × 1, 3 × 1)	(7 × 1, 7 × 1, 5 × 1, 5 × 1, 5 × 1, 5 × 1, 5 × 1, 5 × 1)	(3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1)	(3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1, 3 × 1)
Training time(s)	25.1	32.2	45.7 s	50.5 s	20.1 s
Testing set identification accuracy rate	100.00%	100.00%	98.49%	97.11%	49.49%

Both Structure 1 and Structure 2 identification accuracy rate reached 100.00%, but the training time of Structure 1 was shorter. In the field of image identification, when the same perceptual field of view is reached, the smaller the convolution kernel size, the

smaller the required parameters and calculations, and the shorter training time. Based on the feature extraction of the vibration signal of one-dimensional time series, the faster the training speed will be if the larger convolution kernel size is used. The problem reflected in Structure 3 and Structure 4 is that the structure of the one-dimensional time sequence signal itself is not complicated, and the multi-layer complex network brings varying degrees of over-fitting problems. The results also show that appropriately increasing convolutional kernel size can improve the training speed of Model after analyzing the loss of Structure 5 and the gradient update mode, it is found that a gradient explosion problem has occurred. Gradients can accumulate continuously in the process of network updating and become very large gradients, leading to a large update of the weight value of the network, which makes the network unstable. In extreme cases, the weight value will overflow and cannot be updated. The degradation of the weight matrix results in the reduction of the effective degree of freedom of the model. The contribution of available degrees of freedom of the network to the gradient norm in learning is uneven. With the increase of the number of multiplication matrices (i.e., the depth of network), the matrix product becomes more and more degenerate. In nonlinear networks with hard saturated boundaries (such as ReLU), the degradation process becomes faster and faster as the depth increases. Therefore, Structure 5 is not updated quickly and the accuracy rate is very low.

After analyzing and comparing the training effect many times, the relevant setting and some parameters of the model are as follows:

- (1). Selected ReLU as the activation function after the first pooling layer and the third pooling layer. ReLU has the advantage of making the output of some neurons equal to zero, improving the sparsity of the network, reducing the interdependence of parameters, and alleviating the occurrence of overfitting problems.
- (2). Optimizer selection. After comparing with common optimizers: SGD, BGD, Momentum, NAG, AdaGrad, RMSProp, Adam, it is found that the optimizer RMSProp has the best effect. The learning rate is set to 0.001.
- (3). Add a flatten layer between the third pooling layer and the fully connected layer.
- (4). The dropout provides a computationally inexpensive property but might be a regularization method, which can effectively prevent overfitting. We add dropout between the third pooling layer and the fully connected layer. The dropout rate is set to 0.5.
- (5). The pooling layer is to screen the features in the receptive field and extract the most representative features in the area, which can effectively reduce the output feature scale. It is usually divided into maximum pooling, average pooling, and sum pooling. One of the main reasons for the error of feature extraction in the convolutional neural network is that the parameter error of the convolutional layer causes the deviation of the estimated mean value, and the maximum pooling can effectively reduce such errors. After testing, we found that the maximum pooling effect is best. Optimally, we add a maximum pooling layer after each convolutional layer, the pooling size is 3×1 . The parameter selection is shown in Table 3.

Table 3. The 1D-CNN parameter selection.

Parameter Name	Value
Number of convolutional layers	3
Number of convolutional kernels	(256, 512, 1024)
Convolutional kernel size	$(7 \times 1, 5 \times 1, 5 \times 1)$
Number of pooling layers	3
Pooling size for each layer	$(3 \times 1, 3 \times 1, 3 \times 1)$
Optimizer	RMSProp
Learning rate	0.001
Maximum iterations	100
Batch size	32
Dropout rate	0.5

2.3.2. Experimental Results

The identification accuracy rate of each sample set is shown in Table 4. The identification accuracy rate of training set and validation set varies with the number of iterations as shown in Figure 4. The maximum iteration number is tentatively set to 100, but in order to prevent overfitting, the early-stopping mechanism is introduced in the subsequent training process. In general, the model loss function does not change significantly after 60 iterations, so training is stopped. In order to show the identification accuracy rate of each category of the model in the testing set more clearly, the confusion matrix is introduced in Figure 5. The t-distributed stochastic neighbor embedding (t-SNE) algorithm is used to visualize the features of the last output layer in Figure 6. The experimental results show that the model can identify the ten states in the testing set with 100.00% accuracy rate.

Table 4. Identification accuracy rate of each sample set.

Training Set	Validation Set	Testing Set
100.00%	100.00%	100.00%

Figure 4. Identification accuracy rate curves of the training set and validation set.

Figure 5. Accuracy rate of various types of fault identification in the testing set.

Figure 6. Visual presentation of the last output layer.

2.3.3. Generalization Performance Experiment

Generalization performance is one of the important performance index of neural network model in practical application. In the actual rotating machinery equipment, the bearing load may change at any time. Carried out the generalization performance test of the proposed fault diagnosis model in two methods.

The first method: the model structure and default parameters trained by using vibration data at 2HP load on the drive end were saved. Replacing the data set, with the selected different load data on the drive end (DE) and different load data on the fan end (FE) to train the model, the identification accuracy rate is shown in Table 5, and it can be seen that the identification rate of 100% is maintained under various working conditions.

Table 5. Identification accuracy rate of each sample set in different working conditions.

Working Condition	Training Set	Validation Set	Testing Set
1HP-DE	100.00%	100.00%	100.00%
3HP-DE	100.00%	100.00%	100.00%
1HP-FE	100.00%	100.00%	100.00%
2HP-FE	100.00%	100.00%	100.00%
3HP-FE	100.00%	100.00%	100.00%

The second method: simulating load change. The training set used the load data with the DE 2HP for model training to simulate load changes. The testing set selected ten state corresponding to 1HP and 3HP at the drive end for identification, and the testing set still maintains a 100% accuracy rate. The identification accuracy rate of the testing set is 100.00% and 99.97%, respectively.

2.3.4. Compared with Other Models

At present, many fault diagnosis methods based on CNN can reach a high level of identification accuracy rate for the bearing data set of CWRU. However, the method proposed in this paper has better performance in training time and variable load identification accuracy rate. Compared with the 1D-CNN proposed in this article, and with other models: (1) we selected the method in the paper that also used 1D-CNN [41]; (2) we built a long short-term memory (LSTM) network with the same structure as the paper [42]; and (3) we used MobileNet [43]; and (4) ShuffleNet V2 [43]. The training set used the load data with the DE 2HP. Comparisons with the training time of different models, and the identification accuracy rate of each model on the testing set under different loads, is shown

in Table 6. (“2HP-testing set” means the testing set used the load data with the DE 2HP). The method proposed in this paper used the same training set as the 1D-CNN method of paper [41], and the proposed method has higher identification accuracy rate of the testing set under different loads. The LSTM and CNN principles are different, and the dimensions of the input data of MobileNet and ShuffleNet V2 are different from the proposed method. Therefore, the performance of the proposed method is better only when comparing the identification accuracy rate under different loads.

Table 6. The training time of each model and the identification accuracy rate of the test set under different loads.

	Training Time (s)	2HP-Testing Set	1HP-Testing Set	3HP-Testing Set
Proposed method	25.3	100.00%	100.00%	99.97%
1D-CNN [41]	30.5	100.00%	97.80%	99.56%
LSTM	125.4	100.00%	97.11%	97.69%
MobileNet	988.5	99.49%	95.00%	96.36%
ShuffleNet V2	810.6	99.67%	97.55%	96.34%

3. A Small Sample Size Expansion Method Based on 1D-DCGAN

The amount of data generated in the actual project was very large, but at the same time, within the practical industrial field, there were occasional rotating machinery faults, fault signals were difficult to collect in a timely manner. There was an imbalanced volume of data in the fault diagnosis training model, and the fault samples only accounted for a small part of the data collected. This leads to low accuracy rate in the training model and a poor generalization ability.

In Section 2.3, it is proved that the one-dimensional convolution operation has good feature extraction and expression ability for the vibration signal of the time series, and the one-dimensional convolution operation is further integrated into the GAN. The improved GAN of one-dimensional convolution operation was used to form an expansion method for the small sample size based on one-dimensional deep convolutional generative adversarial networks (1D-DCGAN), while the small sample size fault samples were expanded to build a balanced data set so as to train the fault diagnosis model and improve the identification accuracy rate.

3.1. Generative Adversarial Network

In 2014, Goodfellow et al. proposed the generative adversarial network (GAN), which is a special antagonistic process in which two neural networks compete [44]. The first network generates data, while the second network tries to distinguish real data from fake data created by the first network. The first network is called the generator and denoted by $G(z)$, and the second neural network is called the discriminator and denoted by $D(x)$. The generator $G(z)$ receives the input z from the probability distribution p_z and provides the generated data to the discriminant network $D(x)$. The discriminator network takes real data or generated data as input and tries to predict whether the current input is real data or generated data. One of the inputs x is obtained from the real data distribution p_{data} , and then solve a binary classification problem to produce a scalar value ranging from 0–1. When training, we fix one of them (discrimination network or generation network), update the parameters of the other model, alternate iterations, and reach a Nash equilibrium. Ultimately, the generative model can estimate the distribution of sample data. The generator network takes random noise as input and tries to generate sample data.

The structure of the GAN is shown in Figure 7, and the objective function is:

$$\min_G \max_D V(D, G) = E_{p_{data}(x)}[\log(D(x))] + E_{p_z(z)}[\log(1 - D(G(z)))] \quad (5)$$

Figure 7. Structural diagram of GAN.

The emergence of generative adversarial networks has dramatically elevated the research of unsupervised learning and image generation. At present, it has been extended to all kinds of fields of the computer vision, but there are few researches on processing time series one-dimensional signals. This paper constructs a 1D-DCGAN to generate one-dimensional vibration signals.

3.2. 1D-DCGAN

Deep convolutional generative adversarial networks (DCGAN) are a variant of GAN. Radford et al. used DCGAN for unsupervised representation learning and first mentioned DCGAN [45]. Based on this, this paper makes further improvements to traditional GANs and builds a 1D-DCGAN model, as shown in Figure 8. The 1D-DCGAN model uses some architectural constraints to solidify the network:

- (1) In the generator and discriminator, the convolution operation is one-dimensional convolution, and the deconvolution operation is not used in the generator.
- (2) In the discriminator, the strided convolutions are used to replace the pooling layers, and in the generator, only the convolution operation with a stride length of one is used to replace the pooling layers.
- (3) We eliminate the fully connected layer of the hidden layer in the generator and discriminator.
- (4) In the generator, the activation function Tanh is used in the last fully connected (Fc) layer, and ReLU activation is used in every convolutional layer.
- (5) In the discriminator, LeakyRelu activation is used in all convolutional layers, and the activation function Tanh is used in the last fully connected (Fc) layer.

Figure 8. The 1D-DCGAN structure diagram.

The loss function of the original GAN has defects. The analysis found that the better the discriminator is trained, the more serious the gradient of the generator disappearance, which limits the training of the generator. Martin Arjovsky et al. proposed WGAN [46], which uses Wasserstein distance instead of Jensen–Shannon divergence to avoid a certain gradient disappearance. Ishaan Gulrajani et al. continued to improve on this basis and proposed WGAN-GP [47], and the Lipschitz limit was reflected by an additional loss item $\left[\|\nabla_x D(x)\|_p - K\right]^2$, where K was set to 1.

Generator loss:

$$L1 = -E_{\tilde{x} \sim p_g} [D(\tilde{x})] \quad (6)$$

Discriminator loss:

$$L2 = E_{\tilde{x} \sim p_g} [D(\tilde{x})] - E_{x \sim p_r} [D(x)] + \lambda E_{\hat{x} \sim p_{\hat{x}}} \left[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2 \right] \quad (7)$$

where p_r is the real data distribution, p_g is the data distribution of the generator transform, $\tilde{x} = G(z)$, $z \sim p(z)$, $p(z)$ is the distribution of random noise, $p_{\hat{x}}$ is the random interpolation sampling distribution on the line p_r and p_g , λ is the gradient penalty coefficient.

The specific training process of 1D-DCGAN is described by Algorithm 1.

Algorithm 1. The method in this paper is based on WGAN-GP. Some default values are: learning rate $\alpha = 0.00001$, batch size $a = 128$, $\lambda = 10$.

Require: The number of critic iterations per generator iteration $k_{critic} = 5$

Require: The initial critic parameter ω_0 , initial generator parameter γ_0

while γ has not converged **do**

for $t = 1, \dots, k_{critic}$ **do**

for $j = 1, \dots, a$ **do**

$x \sim p_r, z \sim p(z)$, a random number $\delta \sim U[0, 1]$

$\tilde{x} \leftarrow G_{\gamma}(z)$

$\hat{x} \leftarrow \delta x + (1 - \delta)\tilde{x}$

$L^{(j)} \leftarrow D_{\omega}(\tilde{x}) - D_{\omega}(x) + \lambda(\|\nabla_{\hat{x}} D_{\omega}(\hat{x})\|_2 - 1)^2$

End for

$\omega \leftarrow \text{RMSPProp}(\nabla_{\omega} \frac{1}{a} \sum_{j=1}^a L^{(j)}, \omega, \alpha)$

End for

Sample a batch of latent variables $\{z^{(i)}\}_{i=1}^a \sim p(z)$

$\gamma \leftarrow \text{RMSPProp}(\nabla_{\gamma} \frac{1}{a} \sum_{j=1}^a -D_{\omega}(G_{\gamma}(z)), \gamma, \alpha)$

end while

3.3. Experimental Verification

The test verification is still selected the bearing data set of CWRU, and the sampling frequency was 12 kHz and the load was selected as the drive end vibration data under 2HP. The real sample construction method in the discriminator is consistent with the Section 2.3 above, and normal samples are not expanded. The sample length is 1024, and the number is 400. According to the label, nine kinds of fault signals are input into the 1D-DCGAN model in batches. In the analysis of commonly used DCGAN models, the generator uses the process of inverse convolution and the characteristics of one-dimensional signals. Thus, the length of one-dimensional random noise input in the generation network constructed is 1024, which is consistent with the signal length in the original sample. The generation network does not use the inverse convolution operation, and only performs the convolution operation without changing the dimension. In both the generation network and the discriminant network, the optimizer selected RMSProp, the learning rate is 0.00001, the generator and the discriminator are trained alternately, and the number of iterations is tentatively set to 2×10^6 . Refer to the parameter selection of 1D-CNN in Section 2.3.3, after

multiple models were trained, and we compared the generated signal with the original signal, some parameters of the model determined are shown in Table 7.

Table 7. The 1D-DCGAN parameter selection.

Module	Network Layer	Convolutional Kernel Size	Number of Convolutional Kernels	Stride	Activation Function
Generator	Conv1D_1	5×1	64	1	ReLU
	Conv1D_2	5×1	64	1	ReLU
	Conv1D_3	5×1	64	1	ReLU
	Conv1D_4	5×1	64	1	ReLU
	Conv1D_5	5×1	64	1	ReLU
	Fc				Tanh
Discriminator	Conv1D_1	5×1	64	3	LeakyReLU
	Conv1D_2	5×1	128	5	LeakyReLU
	Conv1D_3	5×1	256	3	LeakyReLU
	Conv1D_4	5×1	512	5	LeakyReLU
	Conv1D_5	5×1	1024	3	LeakyReLU
	Fc				Tanh

We selected the generated data with the Label 2 fault type to explanation. The comparison between a set of original signal and the generated signal are shown in Figure 9. The vibration curves of five samples of generated data with the number of iterations are selected for display as shown in Figure 10. As the number of iterations increases, the generated data is getting closer and closer to the original data.

Figure 9. Vibration curves of real data (blue) and generated data (red).

A key issue was working out when to stop the GAN's training. We saved the loss value every 1000 iterations, and the loss value for generators and discriminators varies with the number of iterations as shown in Figure 11. During the training process, it was found that after multiple iterations, the loss of the network could not be reduced, and it continued oscillating within a small range. It was found that after 2×10^4 iterations of each round of tests, the loss function in the generator and the discriminator no longer changed significantly, but the quality of the generated data was still far below the requirements. The identification accuracy rate of the fault diagnosis model (1D-CNN) trained on the new data generated by selecting the number of iterations of 2×10^5 , 5×10^5 and 10^6 is very low, and the number of iterations is finally determined to be 2×10^6 . From the results of this experiment, it can draw a conclusion that the change to the loss function when the number of iterations is small can be used as a reference condition for model improvement, but that it cannot be used as a condition for stopping the training. The loss value of a well-trained

GAN is always maintained near a value with always fluctuates in a small range. A single GPU, model number is Titan RTX, was used for the training of the 1D-DCGAN model constructed in this article. The training time for each fault sample is approximately 4 hours. Therefore, the model has excellent actual deployment conditions.

Figure 10. The generated signals vary with the number of iterations: (a) 10 iterations; (b) 5×10^3 iterations; (c) 5×10^4 iterations; (d) 8×10^4 iterations; (e) 1.2×10^6 iterations; and (f) 2×10^6 iterations.

Figure 11. Loss of generator and discriminator networks.

Taking a piece of data generated with Label 2 as an example for analysis, five time-domain indicators that can better reflect the characteristics of the signal in the time-domain were selected: kurtosis, peak indicator, margin indicator, waveform indicator, and impulse indicator. The original signal was compared with the generated signal. The values are shown in Table 8. Fast Fourier transform was then performed on the two signals, as shown in Figure 12, and it was observed that the amplitudes of the generated signal and the original signal at different frequencies are basically the same.

Table 8. The characteristic parameters of the original signal and the generated signal.

	Kurtosis	Peak Indicator	Margin Indicator	Waveform Indicator	Impulse Indicator
Original signal	7.376	9.187	20.965	1.629	14.974
Generated signal	6.895	8.928	18.192	1.549	13.525

Figure 12. Contrast the original signal spectrum with the generated signal spectrum: (a) original signal; (b) generated signal.

The training times for the bearing data of the remaining eight types are first tentatively set to 2×10^6 times. Similarly, the time-domain and frequency-domain data analysis of the generated signals showed that the signals generated well after 2×10^6 iterations and restored the characteristics of the original signals to a high level. The effects of the vibration

curve on the remaining eight types of the original signals (blue) and their corresponding generated signals (red) are shown in Figure 13.

Figure 13. Comparison of real data (blue) and generated data (red) for each fault type: (a) Label 1; (b) Label 3; (c) Label 4; (d) Label 5; (e) Label 6; (f) Label 7; (g) Label 8; and (h) Label 9.

3.4. Compared with Other Models

The development of data generation model has experienced from the manual establishment of relevant mathematical model to the current mainstream data generation model using neural network. Since there are few researches on the generation of rolling bearing original signals, two representative data generation methods based on time series are selected in this part. One method is based on probability statistics theory and generates data randomly by transition matrix method of Markov chain process [48]. Another method is the representative data generation method in deep learning: variational auto-encoder (VAE). The paper introduced VAE into fault diagnosis framework to realize data amplification by vibration signal generation [49]. Samples generated by the three methods were tested in two different ways: first, the generated data and the original data were evenly mixed in order to construct new training samples and then, they were inputted into the 1D-CNN fault diagnosis model (Test Method 1). The second method, the generated data was selected as the training set, and the original data as the testing set (Test Method 2). The identification accuracy rate of the three models in the two test methods is shown in Table 9. Taking the generated data of Label 7 bearing as an example, the average value of each characteristic index is also shown in Table 9 (the characteristic index of the original data of Label 7 bearing is Kurtosis: 6.8754; wave-form indicator: 1.4535; skewness: 0.2549.) It can be seen that the data generated by the 1D-DCGAN model can better restore the original data.

Table 9. Performance comparison of the three methods.

	Test Method 1	Test Method 2	Kurtosis	Waveform Indicator	Skewness	Generated Sample
Proposed method	100.00%	99.49%	6.9219	1.4745	0.2623
Markov chain [48]	90.34%	75.56%	2.9308	1.2878	−0.0368
VAE [49]	99.79%	95.49%	5.8629	1.4255	−0.0583

4. Fault Diagnosis Experiment of Hydraulic Pump

4.1. Experiment Introduction

This experiment was completed on a comprehensive test bench for hydraulic pump failure simulation and condition monitoring. The test bench meets the requirements of test verification.

The test took MCY14-1B type axial piston pump as the research object, the component models and performance parameters of the test system are shown in Table 10. We set the system pressure to 10 MPa and installed a vibration acceleration sensor at the end cover of the pump, as shown in Figure 14. The fault samples of the axial piston pump were artificially designed to simulate three failure states, and at the same time, the failure degrees of sliding shoes wear and central spring failure were distinguished. Finally, a total of eight working conditions were set: normal state, swashplate wear, sliding shoes Wear 1 (the wear extent is 1.5 mm), sliding shoes Wear 2 (the wear extent is 2 mm), sliding shoes Wear 3 (the wear extent is 2.5 mm), center spring Failure 1 (the wear extent is 0.6 mm), center spring Failure 2 (the wear extent is 1.0 mm), and center spring Failure 3 (the wear extent is 1.4 mm).

Table 10. Component models and performance parameters.

Component Name	Component Type	Performance Parameter
Driving Motor	Y132M-4	rated speed:1480 rpm
Axial piston pump	MCY14-1B	Displacement:10 mL/r
Data acquisition card	PCI-1742U	Maximum sampling rate: 1 MS/s
Vibration Sensor	YD72D	Frequency range: 1–18,000 Hz

Figure 14. Axial piston pump failure simulation test bench and sensor layout.

We analyzed the collected original vibration signals and simulating the problem of imbalanced sample in actual engineering. The 1D-DCGAN model proposed in this paper is used to expand the collection of small sample size fault samples. We analyzed and compared the unexpanded sample data and the expanded sample data to identify the accuracy rate of hydraulic pump faults. The composition of the two categories samples is shown in Table 11. Each sample type was randomly selected to form a training set, verification set and testing set, with a ratio of 3:1:1.

Table 11. Sample composition in two categories.

Sample Type	Sample Length	Number of Original Samples	Number of Samples after Expansion	Label
Sliding shoes wear 1	2048	1000	1000	0
Sliding shoes wear 2	2048	1000	1000	1
Sliding shoes wear 3	2048	1000	1000	2
Swashplate wear	2048	50	200	3
Center spring failure 1	2048	200	400	4
Center spring failure 2	2048	200	400	5
Center spring failure 3	2048	100	200	6
Normal	2048	1000	1000	7

4.2. Result and Analysis

We used two methods to train the 1D-CNN model. Method 1 is a model trained with imbalanced data that has not been expanded, and Method 2 is a model trained with expanded data. The fault identification accuracy rate of the two methods for the axial piston pump are shown in Table 12. Figure 15 is the confusion matrices of the identification accuracy rate of the various faults of the axial piston pump in the testing set. As a result of Method 1, the sample set is imbalanced, it has a notably impact on the accuracy rate of fault identification, and the effect on the testing set is poor, especially for the fault identification rate of Label 3 is very low. The imbalanced sample set of the training model will bring the following problems: (1) During model training, the feature learning of a small number of fault samples is incomplete, making it misjudged into a certain category. (2) A small number of samples may be randomly divided into validation set and testing set, resulting in the absence of a small number of fault samples in the training set. (3) When the data amount of a single fault type is too small and the network structure is relatively complex, it will cause the model to overfit this fault. In Method 2, the model trained by the expanded sample has significantly improved the accuracy rate of fault identification.

Table 12. The identification accuracy rate of the two methods.

	Training Set	Validation Set	Testing Set
Method 1	95.49%	91.33%	76.47%
Method 2	99.96%	99.87%	98.01%

Figure 15. The identification accuracy rate of the two methods in the testing set: (a) Method 1; and (b) Method 2.

5. Conclusions

This paper proposes a complete solution to the two major problems of the difficulty of feature extraction from massive data and the small sample size of fault samples in the fault diagnosis of rotating machinery.

The 1D-CNN intelligent fault diagnosis method proposed in this paper takes the original vibration signal of the rotating mechanical element as the input of the model, performs feature extraction in the convolutional layers, and reduces feature dimensionality in the pooling layers, which realize adaptive feature extraction and dimensionality reduction, and the output of the network is the diagnosis result. The experiment proves that the fault diagnosis of the bearing and axial piston pump is very accurate. Moreover, it has good robustness and generalization performance, and maintains high identification accuracy

rate even with load changes. The model can accurately extract the common features of the same fault type signals with dissimilar loads. It is found in the experiment that appropriately increased the size of one-dimensional convolution kernel can improve the processing efficiency of one-dimensional time-series data.

The proposed expansion method based on the small sample data of 1D-DCGAN generates small sample size fault vibration signals of rotating mechanical components with high quality, which effectively solves the problem of imbalanced sample during model training. There are three important ideas. Firstly, according to the characteristics of one-dimensional signal, the dimension characteristics of the random noise in the traditional generative adversarial network are improved. The random noise is consistent with the real sample dimension. Secondly, instead of using the inverse convolution operation in the generating network, the convolution operation with stride Size 1 is used to replace pooling layer. In the end, gradient penalty and Wasserstein distance are applied to the training process of the 1D-DCGAN. These ideas improve the stability of the model and reduce the training time. The accuracy rate of axial piston pump fault diagnosis model based on balanced data training has been greatly improved.

The proposed 1D-CNN and 1D-DCGAN provide a useful reference scheme for processing one-dimensional signals of time series relying on deep learning method, which can effectively improve the fault diagnosis effect of rotating mechanical components based on vibration signals.

Author Contributions: Formal analysis, C.W.; funding acquisition, W.J.; investigation, S.Z.; methodology, C.W.; project administration, W.J.; resources, C.W.; software, C.W. and J.Z.; validation, C.W. and J.Z.; writing—original draft, C.W.; writing—review and editing, W.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China (Grant No. 51875498, 51475405), and the Key Project of Natural Science Foundation of Hebei Province, China (Grant No. E2018203339, F2020203058).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments: This work is supported by National Natural Science Foundation of China (Grant No. 51875498, 51475405), and the Key Project of Natural Science Foundation of Hebei Province, China (Grant No. E2018203339, F2020203058). The support is gratefully acknowledged. The authors would also like to thank the reviewers for their valuable suggestions and comments.

Conflicts of Interest: The authors declare that there are no conflict of interest regarding the publication of this paper.

References

1. Sahal, R.; Breslin, J.G.; Ali, M.I. Big data and stream processing platforms for Industry 4.0 requirements mapping for a predictive maintenance use case. *J. Manuf. Syst.* **2020**, *54*, 138–151. [\[CrossRef\]](#)
2. Yin, S.; Kaynak, O. Big Data for Modern Industry: Challenges and Trends [Point of View]. *Proc. IEEE* **2015**, *103*, 143–146. [\[CrossRef\]](#)
3. Sekhar, A. Multiple cracks effects and identification. *Mech. Syst. Signal Process.* **2008**, *22*, 845–878. [\[CrossRef\]](#)
4. Gasch, R. Dynamic behaviour of the Laval rotor with a transverse crack. *Mech. Syst. Signal Process.* **2008**, *22*, 790–804. [\[CrossRef\]](#)
5. Combet, F.; Gelman, L. Optimal filtering of gear signals for early damage detection based on the spectral kurtosis. *Mech. Syst. Signal Process.* **2009**, *23*, 652–668. [\[CrossRef\]](#)
6. Ming, A.; Zhang, W.; Qin, Z.; Chu, F. Envelope calculation of the multi-component signal and its application to the deterministic component cancellation in bearing fault diagnosis. *Mech. Syst. Signal Process.* **2015**, *50–51*, 70–100. [\[CrossRef\]](#)
7. Xie, H.; Lin, J.; Lei, Y.; Liao, Y. Fast-varying AM-FM components extraction based on an adaptive STFT. *Digit. Signal Process.* **2012**, *22*, 664–670. [\[CrossRef\]](#)
8. Tang, B.; Liu, W.; Song, T. Wind turbine fault diagnosis based on Morlet wavelet transformation and Wigner-Ville distribution. *Renew. Energy* **2010**, *35*, 2862–2866. [\[CrossRef\]](#)

9. Yan, R.; Gao, R.X.; Chen, X. Wavelets for fault diagnosis of rotary machines: A review with applications. *Signal Process.* **2014**, *96*, 1–15. [[CrossRef](#)]
10. Liu, R.; Yang, B.; Zio, E.; Chen, X. Artificial intelligence for fault diagnosis of rotating machinery: A review. *Mech. Syst. Signal Process.* **2018**, *108*, 33–47. [[CrossRef](#)]
11. Fang, W.; Ding, Y.; Zhang, F.; Sheng, J. Gesture Recognition Based on CNN and DCGAN for Calculation and Text Output. *IEEE Access* **2019**, *7*, 28230–28237. [[CrossRef](#)]
12. Mustaqem; Kwon, S. A CNN-Assisted Enhanced Audio Signal Processing for Speech Emotion Recognition. *Sensors* **2019**, *20*, 183. [[CrossRef](#)]
13. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)] [[PubMed](#)]
14. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2012**, *60*, 1097–1105. [[CrossRef](#)]
15. Dong, M.; Fang, Z.; Li, Y.; Bi, S.; Chen, J. AR3D: Attention Residual 3D Network for Human Action Recognition. *Sensors* **2021**, *21*, 1656. [[CrossRef](#)] [[PubMed](#)]
16. Tang, T.; Hu, T.; Chen, M.; Lin, R.; Chen, G. A deep convolutional neural network approach with information fusion for bearing fault diagnosis under different working conditions. *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* **2020**. [[CrossRef](#)]
17. Liu, S.; Xie, J.; Shen, C.; Shang, X.; Wang, D.; Zhu, Z. Bearing Fault Diagnosis Based on Improved Convolutional Deep Belief Network. *Appl. Sci.* **2020**, *10*, 6359. [[CrossRef](#)]
18. Shao, S.; Wang, P.; Yan, R. Generative adversarial networks for data augmentation in machine fault diagnosis. *Comput. Ind.* **2019**, *106*, 85–93. [[CrossRef](#)]
19. Kong, X.; Mao, G.; Wang, Q.; Ma, H.; Yang, W. A multi-ensemble method based on deep auto-encoders for fault diagnosis of rolling bearings. *Measurement* **2020**, *151*, 107132. [[CrossRef](#)]
20. Luo, J.; Huang, J.; Li, H. A case study of conditional deep convolutional generative adversarial networks in machine fault diagnosis. *J. Intell. Manuf.* **2021**, *32*, 407–425. [[CrossRef](#)]
21. Xu, Z.; Li, C.; Yang, Y. Fault diagnosis of rolling bearings using an Improved Multi-Scale Convolutional Neural Network with Feature Attention mechanism. *ISA Trans.* **2021**, *110*, 379–393. [[CrossRef](#)] [[PubMed](#)]
22. Tang, S.; Yuan, S.; Zhu, Y. Data Preprocessing Techniques in Convolutional Neural Network Based on Fault Diagnosis Towards Rotating Machinery. *IEEE Access* **2020**, *8*, 149487–149496. [[CrossRef](#)]
23. Qiao, M.; Yan, S.; Tang, X.; Xu, C. Deep Convolutional and LSTM Recurrent Neural Networks for Rolling Bearing Fault Diagnosis Under Strong Noises and Variable Loads. *IEEE Access* **2020**, *8*, 66257–66269. [[CrossRef](#)]
24. Wan, L.; Chen, Y.; Li, H.; Li, C. Rolling-Element Bearing Fault Diagnosis Using Improved LeNet-5 Network. *Sensors* **2020**, *20*, 1693. [[CrossRef](#)] [[PubMed](#)]
25. Tang, S.; Zhu, Y.; Yuan, S.; Li, G. Intelligent Diagnosis towards Hydraulic Axial Piston Pump Using a Novel Integrated CNN Model. *Sensors* **2020**, *20*, 7152. [[CrossRef](#)] [[PubMed](#)]
26. Tang, S.; Yuan, S.; Zhu, Y. Convolutional Neural Network in Intelligent Fault Diagnosis Toward Rotatory Machinery. *IEEE Access* **2020**, *8*, 86510–86519. [[CrossRef](#)]
27. Hoang, D.-T.; Kang, H.-J. Rolling element bearing fault diagnosis using convolutional neural network and vibration image. *Cogn. Syst. Res.* **2019**, *53*, 42–50. [[CrossRef](#)]
28. Liu, Y.; Yang, Y.; Feng, T.; Sun, Y.; Zhang, X. Research on Rotating Machinery Fault Diagnosis Method Based on Energy Spectrum Matrix and Adaptive Convolutional Neural Network. *Processes* **2020**, *9*, 69. [[CrossRef](#)]
29. Tang, X.; He, Q.; Gu, X.; Li, C.; Zhang, H.; Lu, J. A Novel Bearing Fault Diagnosis Method Based on GL-mRMR-SVM. *Processes* **2020**, *8*, 784. [[CrossRef](#)]
30. Zhao, B.; Zhang, X.; Li, H.; Yang, Z. Intelligent fault diagnosis of rolling bearings based on normalized CNN considering data imbalance and variable working conditions. *Knowl. Based Syst.* **2020**, *199*, 105971. [[CrossRef](#)]
31. Zhou, Q.; Li, Y.; Tian, Y.; Jiang, L. A novel method based on nonlinear auto-regression neural network and convolutional neural network for imbalanced fault diagnosis of rotating machinery. *Measurement* **2020**, *161*, 107880. [[CrossRef](#)]
32. Wei, J.; Huang, H.; Yao, L.; Hu, Y.; Fan, Q.; Huang, D. New imbalanced fault diagnosis framework based on Cluster-MWMOTE and MFO-optimized LS-SVM using limited and complex bearing data. *Eng. Appl. Artif. Intell.* **2020**, *96*, 103966. [[CrossRef](#)]
33. Mao, W.; He, L.; Yan, Y.; Wang, J. Online sequential prediction of bearings imbalanced fault diagnosis by extreme learning machine. *Mech. Syst. Signal Process.* **2017**, *83*, 450–473. [[CrossRef](#)]
34. Viola, J.; Chen, Y.; Wang, J. FaultFace: Deep Convolutional Generative Adversarial Network (DCGAN) based Ball-Bearing failure detection method. *Inf. Sci.* **2021**, *542*, 195–211. [[CrossRef](#)]
35. Mariani, G.; Scheidegger, F.; Istrate, R.; Bekas, C.; Malossi, C. BAGAN: Data Augmentation with Balancing GAN. *arXiv* **2018**, arXiv:1803.09655.
36. Li, Q.; Chen, L.; Shen, C.; Yang, B.; Zhu, Z. Enhanced generative adversarial networks for fault diagnosis of rotating machinery with imbalanced data. *Meas. Sci. Technol.* **2019**, *30*, 115005. [[CrossRef](#)]
37. Mao, W.; Liu, Y.; Ding, L.; Li, Y. Imbalanced Fault Diagnosis of Rolling Bearing Based on Generative Adversarial Network: A Comparative Study. *IEEE Access* **2019**, *7*, 9515–9530. [[CrossRef](#)]
38. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.

-
39. Lee, C.-Y.; Gallagher, P.W.; Tu, Z. Generalizing Pooling Functions in Convolutional Neural Networks: Mixed, Gated, and Tree. In Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, Cadiz, Spain, 9–11 May 2016; Volume 51, pp. 464–472.
 40. Liu, W.; Wen, Y.; Yu, Z.; Yang, M. Large-Margin Softmax Loss for Convolutional Neural Networks. *arXiv* **2016**, arXiv:1612.02295.
 41. Chen, C.-C.; Liu, Z.; Yang, G.; Wu, C.-C.; Ye, Q. An Improved Fault Diagnosis Using 1D-Convolutional Neural Network Model. *Electronics* **2020**, *10*, 59. [[CrossRef](#)]
 42. An, Z.; Li, S.; Wang, J.; Jiang, X. A novel bearing intelligent fault diagnosis framework under time-varying working conditions using recurrent neural network. *ISA Trans.* **2020**, *100*, 155–170. [[CrossRef](#)]
 43. Liu, H.; Yao, D.; Yang, J.; Li, X. Lightweight Convolutional Neural Network and Its Application in Rolling Bearing Fault Diagnosis under Variable Working Conditions. *Sensors* **2019**, *19*, 4827. [[CrossRef](#)] [[PubMed](#)]
 44. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Qzair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *arXiv* **2014**, arXiv:1406.2661. [[CrossRef](#)]
 45. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. In Proceedings of the 4th International Conference on Learning Representations, ICLR 2016—Conference Track Proceedings, San Juan, PR, USA, 2–4 May 2016.
 46. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein GAN. *arXiv* **2017**, arXiv:1701.07875.
 47. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A. Improved training of Wasserstein GANs. *arXiv* **2017**, arXiv:1704.00028.
 48. Shamshad, A.; Bawadi, M.; Wanhussin, W.; Majid, T.; Sanusi, S. First and second order Markov chain models for synthetic generation of wind speed time series. *Energy* **2005**, *30*, 693–708. [[CrossRef](#)]
 49. Zhao, D.; Liu, S.; Gu, D.; Sun, X.; Wang, L.; Wei, Y.; Zhang, H. Enhanced data-driven fault diagnosis for machines with small and unbalanced data based on variational auto-encoder. *Meas. Sci. Technol.* **2019**, *31*, 035004. [[CrossRef](#)]