Bearing Fault Diagnosis Based on a Novel Adaptive ADSD-gcForest Model

Shuo Zhai, Zhenghua Wang and Dong Gao *

School of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China; shzhaishuo@163.com (S.Z.); 2019210490@mail.buct.edu.cn (Z.W.)
*
Correspondence: gaodong@mail.buct.edu.cn

Abstract: With the continuous improvement of industrial production requirements, bearings work significantly under strong noise interference, which makes it difficult to extract fault features. Deep Learning-based approaches are promising for bearing diagnosis. They can extract fault information efficiently and conduct accurate diagnosis. However, the structure of deep learning is often determined by trial and error, which is time-consuming and lacks theoretical support. To address the above problems, an adaptive (Adaptive Depthwise Separable Dilated Convolution and multi-grained cascade forest) ADSD-gcForest fault diagnosis model is proposed in this paper. Multiscale convolution combined with convolutional attention mechanism (CBAM) concentrates on effectively extracting fault information under strong noise, and the Meta-Activate or Not (Meta-ACON) activation function is integrated to adaptively optimize the model structure according to the characteristics of input samples, then gcForest outputs the final diagnosis result as the classifier. The experiment compares the effects of three bearings failure diagnoses under various noise and load conditions. The experimental results show the effectiveness and practicability of the proposed method.

Keywords: fault diagnosis; Meta-ACON; ADSD-gcForest; SDPimage

1. Introduction

With the development of the manufacturing industry, rolling bearings, as one of the core components of mechanical equipment, play an increasingly irreplaceable role. However, under the condition of strong noise and multiple loads for a long time, the bearings are prone to wear or breakage. An expected failure, such as a crack in the bearings, may cause the breakdown of the entire machine, resulting in magnificent economic loss and severe safety accidents [1]. Therefore, it is of great significance to realize the high efficiency and accuracy of bearing fault diagnosis.

The location of the bearing failure is generally located in the inner ring, outer ring, and rolling element, and the bearing fault usually produces periodic vibrations when machinery is running, so analysis of the vibration signal during bearing operation is the key to achieving the diagnosis of the fault [2]. Traditional fault diagnosis methods are divided mainly into linear and non-linear methods. Linear diagnosis methods mainly contain time domain analysis, frequency domain analysis, and time-frequency analysis [3]. Nonlinear analysis is less adopted in fault diagnosis than linear analysis, chaos theory, fractal dimension, and entropy value theory, are commonly applied nonlinear analysis methods, among others. However, due to the increase in bearing fault datasets and the increasing complexity of production environments, traditional fault diagnosis methods that rely on traditional manual fault sign extraction have become no longer applicable [4]. Therefore, constructing novel fault diagnosis models based on approaches of deep learning have become a research hotspot.

Frequently used deep learning models include the Deep Autoencoder (DAE), the Deep Belief Network (DBN), the Convolutional Neural Network (CNN) and the Recurrent
Neural Network (RNN). Among them, the improved Stack Denoising Autoencoder (SDAE) diagnostic method was proposed by Hou et al. [5], in which the hyperparameters of the DAE network were adaptively selected by the particle swarm algorithm to determine the structure of the SDAE network. On this basis, the characteristic representation of the fault state was obtained, which was input into the Softmax classifier for fault classification and recognition; this method has achieved accurate fault diagnosis under the circumstance of variable operating conditions. In-depth research on DAE was conducted by Shao et al. [6], in which DAE and shrinking auto-encoding were introduced to improve the fault extraction capabilities of faulty features, and local preservation of projection fusion features was applied to optimize feature quality. Liang T et al. [7] presented a method for the diagnosis of rolling bearing failures, which consisted mainly of three steps: a series of DBNs with different hyperparameters were constructed and trained, after which the improved ensemble method was applied to acquire the weight matrix for each DBN and then each DBN voted together according to its respective weight matrix to obtain the final result of the diagnosis. The method of DBN-based degradation assessment under accelerated life testing of bearings was adopted by Ma et al. [8]. Shao et al. proposed a DBN for the diagnosis of induction motors faults, in which vibration signals were introduced directly as input [9], and the t-SNE algorithm was adopted to visualize the learning representation. Han Tao et al. [10] used CNN training to obtain the corresponding characteristic diagram of the multi-wavelet coefficient branching process through the wavelet transform to realize the intelligent diagnosis of rolling bearing composite faults. Liang et al. [11] have constructed two different CNNs, one for extracting time-domain features and the other was applied to extract time-frequency domain features, and then fused them with the time-frequency features and time-domain features extracted by continuous wavelet transform to diagnose faults of rolling bearings in a characteristic way. Bearing fault diagnosis based on LSTM (Long Short-Term Network) and CNN models was established by Pan et al. [12], a fault diagnosis method was proposed by Zhang et al. [13], in which self-encoding of convolutional noise reduction was performed to achieve feature extraction and CNN was introduced for pattern recognition. The long- and short-term memory stacking network was designed by Yu et al. [14], where 12 different bearing health conditions were classified using augmented data, including the type and severity of bearing failure. A convolutional bidirectional long- and short-term memory network was designed by Zhao et al. [15] for bearing fault diagnosis. In this method, a convolutional neural network was applied as a feature extractor of the original signal, and then the bearing faults were classified through a bidirectional long and short-term memory network.

Based on the brief review of the existing diagnosis approaches, the challenges can be summarized as follows: first of all, numerous methods only conduct comparative experiments for a single type of noise and other types of noise are not considered. Second, deep learning structures are often determined by trial and error, which means this structure is randomly defined; as a result, the model with the best performance is adopted after many experiments [16]. To solve the above problem, an adaptive ADSD-gcForest fault diagnosis model is proposed in this paper, and based on the basis of the existing traditional network, the core fault features at different scales are effectively extracted by using dilated convolution with different dilation rates and CBAM fusion under strong noise interference. On this basis, deep separable convolution is incorporated into the dilated convolution mechanism to improve the efficiency of the calculation [17]. In recent years, many adaptive optimization methods have been developed for network structure, but most of these approaches require the assistant of an intelligent optimization algorithm or migration learning [18,19]. In contrast, the network structure can be simply optimized by the MetaACON activation based on the input samples without the need for additional complex algorithms and can not only optimize the model structure but also make the model better deal with different sample data. Then, through the multigranular scanning of the deep forest classifier and the cascade forest algorithm, the hidden fault features in the feature
vector are analyzed and extracted, and the final classification results are output. The main contributions of this article are as follows:

1. An adaptive ADSD-gcForest diagnostic model is proposed for the diagnosis of rolling bearing fault diagnosis, allowing the extraction of features under the high-noise and complex working conditions that could be realized. The structure of the diagnosis model achieves adaptive optimization based on the characteristics of the sample data.

2. Combining the multiscale depth-separable dilated convolution with CBAM can effectively extract fault features under strong noise interference. On the basis of the lack of adjust the original structure of the model, the Meta-ACON activation function is introduced into the convolution layer of the model to achieve adaptive optimization of the model structure according to the fault data of different bearings.

3. The comparative experiment shows that the ADSD-gcForest model proposed in this paper has strong generalization ability and robustness with certain practical value.

The rest of this paper is organized in the following way: the introduction of the related theories is mainly in Section 2, the specific structure of the adaptive ADSD-gcForest model is described in Section 3, Section 4 is the experimental comparison part, and the conclusion is drawn in Section 5.

2. Related Works

2.1. SDP Image

Through the normalization method, the time domain signal can be described in the polar coordinate system. Thus, the vibration signal can be converted to SDP images [20], and the relationship between the amplitude and the frequency of the vibration signal can be described simply and directly through the geometric shape. The specific mapping relationship is as follows:

\[ r(i) = \frac{x_i - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} \]  

\[ \theta(i) = \theta_l + \frac{x_{i+a} - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} \delta \]  

\[ \varphi(i) = \theta_l - \frac{x_{i+a} - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}} \delta \]  

where the input vibration signal is represented as \( x_i \), \( i \) represents the sequence number of the discrete sampling point of the signal in the time domain, the maximum and minimum values of the vibration signal are, respectively, described as \( x_{\text{max}} \) and \( x_{\text{min}} \) and the amplitude of \( x_i \) corresponding to the time lag coefficient \( a \) is shown as \( x_{i+a} \), the radius of the polar coordinates is indicated as \( r(i) \), \( \delta \) is the magnification angle, \( \theta_l \) is the angle of the \( l \)-th mirror symmetry plane, \( \theta(i) \) and \( \varphi(i) \) are the deflection angles of the mirror symmetry plane, where \( \delta \leq \theta_l \) and \( \theta_l = \frac{360}{\lambda} \) (\( l \in (0, N-1) \)), and \( N \) is the number of symmetry planes. SDP images of different fault characteristics will present various geometric characteristics, which are manifested mainly in the curvature, thickness, geometric center and concentrated area of the image arm of the SDP image [21]. The SDP images of different fault characteristics are shown in Figure 1. Among them, IR, OR and B, respectively, represent the inner ring, outer ring and rolling element, while 007, 014, and 021 indicate that the fault diameter is 0.1778 mm, 0.3556 mm and 0.5334 mm separately.
2.2. Dilated Convolutional

Unlike ordinary convolution, the dilated convolution is one of the convolutional neural networks which increases the receptive field of the output unit without increasing the parameters, which is mainly implemented by introducing the dilation rate parameter. Specifically, it mainly depends on the way of interval sampling, which means that the spacing of each value is defined by the dilation rate when the convolution kernel processes the data. Thus, the receptive field can be increased without reducing the image resolution. For convolution kernels of the same size, the larger the dilation rate, the greater the receptive field of the convolution kernel [22]. The calculation formula for the receptive field of the dilated convolution is as follows:

$$r_n = r_{n-1} - 1 + k$$  \hspace{1cm} (4)

where the receptive field of the current layer is presented as $r_n$, $r_{n-1}$ is the upper receptive field and $k$ is the size of the convolution kernel. The sampling process is shown in Figure 2.

![Figure 2. Dilated convolution.](image)

2.3. Depth-Separable Convolution

The standard convolution is decomposed into deep convolution and point-wise convolution by depth-separable convolution [23–25]. First, each channel of the input sample is convolved one by one by the deep convolution; thus, the number of feature maps generated...
is the same as the number of channels of the input sample, and after that, the feature map is reconstructed with weights which are assigned on the basis of the designed algorithm according to point-by-point convolution. In this way, the amount of calculation and parameters of the model can be reduced in both time and space. For example, the dimension of the input samples is set as $D_{\text{in}1}, D_{\text{in}2}, C_{\text{in}}$, the output dimension is arranged accordingly as $D_{\text{out}1}, D_{\text{out}2}, C_{\text{out}}$ and the sizes of the convolution kernel are $D_{k1}$ and $D_{k2}$, where $C_{\text{out}}, C_{\text{in}}$ are the number of channels. The formula to calculate the total number of parameters for ordinary convolution and deeply separable convolution is shown in Formulas (5) and (6), respectively, where $C_{\text{conv}}$ represents the total number of parameters for ordinary convolution and the total number of parameters for deeply separable convolution is $C_{\text{separableconv}}$.

$$C_{\text{conv}} = D_{\text{out}1} \times D_{\text{out}2} \times D_{k1} \times D_{k2} \times C_{\text{out}} \times C_{\text{in}}$$

(5)

$$C_{\text{separableconv}} = D_{\text{out}1} \times D_{\text{out}2} \times D_{k1} \times D_{k2} \times C_{\text{in}} + D_{\text{out}1} \times D_{\text{out}2} \times C_{\text{out}} \times C_{\text{in}}$$

(6)

![Figure 3. Depth-separable convolution.](image)

2.4. CBAM

The attention mechanism is derived from the human visual mechanism, which is widely used in image processing. CBAM has been widely used in target detection by skillfully integrating spatial attention mechanism and channel attention mechanism [26–28]. Primarily, the channel attention mechanism selects which features are the key features, and then uses the spatial attention mechanism to learn the location of the key features, strengthening the extraction of the core features of the input sample; from this, in addition, the model can achieve adaptive refinement of the core features in the images. The specific composition structure is shown in Figure 4, where Avgpooling and Maxpooling, respectively, represent the average pooling and maximum pooling, while shared FC means the shared full connectivity layer.
Different from the traditional Softmax classifier, the hidden features of the input feature vector can be learned by gcForest through the superposition of multi-layer random forests, which then output the final classification results [29,30]. It has been proven that the accuracy of the deep forest classifier is about 1–4% higher than that of the Softmax classifier. The deep forest classifier mainly consists of two parts: multi-granularity scanning and cascade forest. The feature vector is sampled in a sliding window to form a new feature vector, which will be input into the cascade forest. After passing through the multi-layer random forest, the final output class probability distribution vector is taken as the final classification result. The specific structure of the deep forest classifier is shown in Figure 5, where K is the dimension of the input vector table type, n is the dimension of the sliding window, m as the category of the classification number and P is the final output vector whose dimension. Furthermore, in Figure 5, the scanned vector is input into the cascade forest, the blue and green two color Forests, respectively, represent random forest and completely random forest, each layer contains two random forests and two completely random forests and each forest after the completion of a training will be an output vector, of which the dimension is C. The output vectors of the four forests are stacked with the output vectors of multi-granularity scanning, and the vector with dimension (C×4+P) is the output; moreover, after multi-layer learning, the output vectors of the last layer are averaged to obtain the final probability category vector with dimension C and the maximum probability of the vector is taken as the classification result.
3. Method

The ADSD-gcForest model will be described in detail in this section. The detailed implementations of the method are described in the following three steps.

Step 1: A sliding window is used to sample vibration signals, then the noise of different intensity is added and the signals are converted into SDP image, and then the sample data are divided into a training set and a test set.

Step 2: The training set is entered into the adaptive ADSD-gcForest model for training and the Meta-ACON activation function is applied to adaptively adjust the network structure, according to different types of sample data to obtain the current optimal model structure, after which the trained model is saved.

Step 3: The trained model is used to directly extract fault features from new images, which results in the final diagnosis. The overall flowchart of the fault diagnosis is drawn in Figure 6.

![Figure 6](image)

**Figure 6.** The overall flowchart of the fault diagnosis.

3.1. Meta-ACON

In order to achieve more effective fault diagnosis based on different bearing fault data, it may be necessary to continuously adjust the existing structure to achieve higher accuracy. In order to solve the above problems, a relatively simple way to achieve adaptive adjustment of the network model is proposed in this paper: by setting a single conversion factor $\beta$, the Meta-ACON activation function can simply select whether to activate the neurons in this layer according to different sample data (activation represents nonlinear output, while on the contrary, non-activation represents linear output). The design of the Meta-ACON activation function is derived from the smooth maximum function, and its formula is as follows:

$$S_\beta(x_1, \ldots, x_n) = \frac{\sum_{i=1}^{n} x_i e^{\beta x_i}}{\sum_{i=1}^{n} e^{\beta x_i}}$$

(7)

where $x_i$ represents the input signal sequence, $\beta$ is the conversion factor, when $\beta \to \infty$, $S_\beta \to \max$ and $\beta \to 0$, and $S_0$ is the arithmetic mean value. Many common activation functions have the form $\max(\eta_a(x), \eta_b(x))$. $\eta_a(x)$ and $\eta_b(x)$ are two freely configurable functions. For example, in the ReLU function, $\eta_a(x) = x$ and $\eta_b(x) = 0$, many activation functions can be expressed in the form of $\max(\eta_a(x), \eta_b(x))$. To simplify the design, only two variables are...
considered here, and the sigmoid function is simplified as $\sigma$. At this time, the approximate relationship is represented as:

$$S_\beta(\eta_a(x), \eta_b(x)) = \eta_a(x) * \frac{e^{\beta \eta_a(x)}}{e^{\beta \eta_a(x)} + e^{\beta \eta_b(x)}} + \eta_b(x) * \frac{e^{\beta \eta_b(x)}}{e^{\beta \eta_a(x)} + e^{\beta \eta_b(x)}}$$

$$= \eta_a(x) * \frac{1}{1 + e^{-\beta(\eta_a(x) - \eta_b(x))}} + \eta_b(x) * \frac{1}{1 + e^{-\beta(\eta_b(x) - \eta_a(x))}}$$

(8)

Furthermore, $\eta_a(x) = p_1 x$, $\eta_b(x) = p_2 x$ and $p_1 x \neq p_2 x$. The Meta-ACON activation function is as follows:

$$S_\beta = (p_1 - p_2) x * \sigma[\beta(p_1 - p_2)x] + p_2 x$$

(9)

Among them, $p_1$ and $p_2$ are two random trainable parameters; therefore, the activation of neurons in this layer can be easily controlled by means of conversion factor $\beta$, where $\beta = \sigma W_1 W_2 \sum_{h=1}^{H} \sum_{w=1}^{W} x_{c,h,w}$, the input sample data is represented as $x_{c,h,w}$ and $c$, $h$ and $w$, respectively, describe the number of channels, width and height of the input sample data. $W_1$ is the convolution of the sample data with the number of input channels as the width of the sample, the number of output channels as the width/r (r is a constant, generally taken as 16) and the convolution core size of $1 \times 1$. Similarly, $W_2$ is also obtained by the convolution with the convolution core size of $1 \times 1$, except that the number of output channels and input channels of convolution are opposite to the setting of $W_1$. Since the $\beta$ value is directly determined by the structural characteristics of the sample data, different sample data will produce different $\beta$ values. Therefore, after many times of training, with the continuous updating of Meta-ACON parameters, the structure of the model can be continuously optimized. The specific calculation process is shown in Figure 7.

![Figure 7. The specific calculation process of Meta-ACON.](image-url)
3.2. ADSD-gcForest

Compared with time-domain signals, SDP images can represent different fault types in a more intuitive and simple way by presenting different geometric features. Therefore, the key to achieve an accurate fault diagnosis is to design a diagnosis model that can effectively extract geometric features from images. The visual geometry group 16 (VGG16) is one of the commonly used models in image processing. Feature extraction is effectively realized by stacking multilayer convolution, and network parameters are reduced by pooling layer. The model in this paper takes VGG16 as the basic framework. However, the structure of VGG16 network is relatively simple. Firstly, although the network is deep, ordinary convolution is widely used in convolution layers, which cannot extract the sample feature information in multiple scales, which limits the feature extraction ability of the network under the intervention of strong noise. Second, most of the activation functions in the convolution layer simply make the input signal become non-linear, so the network does not have good migration learning ability; thus, in the face of different sample data, the performance of the network will become unstable. Moreover, most diagnostic models use Softmax as the final classifier, however, Softmax is not an advanced classifier and cannot learn the feature information that has not been extracted, so as to reduce the final accuracy.

In response to the above problems, the ADSD-gcForest model proposed in this paper makes the following improvements.

Due to the large number of input sample types, in order to increase the model feature extraction range and enrich the feature information, the characteristics of the receptive field are expanded by using the dilated convolution and combined with the residual network to build three branches. Therefore, the construction of three kinds of dilated convolutions with different dilation rates is connected through the residual network, the dilation rate is set to 1, 2 and 3 and the size of the convolution kernel is $3 \times 3$. Thus, multi-scale feature extraction is realized. After the dilated convolution with different expansion rates, it is combined with the CBAM, and the channel attention mechanism is used to measure the importance of different kinds of channel feature information in the feature map at different scales, so as to determine the key points under different channels in the feature map features. Then, the spatial attention mechanism is introduced to locate these key features and extract the key feature information from the feature map to obtain key features at different scales. Next, three feature maps are obtained and integrated using the residual network and input into the next layer. Due to the use of more dilated convolution and attention mechanisms in the network, it may lead to a longer network training time. Since the convolution operations for different channels of the input image can be simultaneously performed by the depth-separable convolution, the depth-separable convolution mechanism is led into the dilated convolution layer, after which the weight ratio of each feature map is determined through quasi-point convolution, and the feature maps are integrated according to the weights. Thus, computational efficiency could be improved in this way. In order to realize that the network model can be adaptively adjusted according to the sample data of different fault types, the original ReLU activation function in the convolutional layer is replaced with the Meta-ACON activation function. The Meta-ACON activation function can be based on the size characteristics of the input image. By setting the conversion factor $\beta$, the value of $\beta$ determines whether to activate the neurons in this layer after multiple trainings, and a flexible network structure can be adopted for the training model according to different input samples. Softmax is replaced by gcForest, which learns the hidden fault characteristics and gives the final results of the diagnosis results. The structure of the model is shown in Figure 8. SD convolution stands for dilated convolution with a deep separable mechanism. Detailed parameters of the optimized network are shown in Table 1.
Table 1. Detailed parameters of the optimized network.

| Layers                  | Filters | Kernel_Size/Dilation Rate | Trainable Parameters | Input_Shape          | Output_Shape          |
|-------------------------|---------|----------------------------|----------------------|----------------------|----------------------|
| separable_conv_1        | 64      | 3/1                        | 137                  | 28 x 28 x 1          | 28 x 28 x 64         |
| CBAM_1                  |         |                            | 677                  | 28 x 28 x 64         | 28 x 28 x 64         |
| separable_conv_2        | 64      | 3/2                        | 137                  | 28 x 28 x 1          | 28 x 28 x 64         |
| CBAM_2                  |         |                            | 677                  | 28 x 28 x 64         | 28 x 28 x 64         |
| separable_conv_3        | 64      | 3/3                        | 137                  | 28 x 28 x 1          | 28 x 28 x 64         |
| CBAM_3                  |         |                            | 677                  | 28 x 28 x 64         | 28 x 28 x 64         |
| Add_1                   | 0       |                            |                      | 28 x 28 x 64         | 28 x 28 x 64         |
| separable_conv_4        | 128     | 3/1                        | 8896                 | 28 x 28 x 128        | 28 x 28 x 128        |
| CBAM_4                  |         |                            | 2277                 | 28 x 28 x 128        | 28 x 28 x 128        |
| separable_conv_5        | 128     | 3/2                        | 8896                 | 28 x 28 x 128        | 28 x 28 x 128        |
| CBAM_5                  |         |                            | 2277                 | 28 x 28 x 128        | 28 x 28 x 128        |
| separable_conv_6        | 128     | 3/3                        | 8896                 | 28 x 28 x 128        | 28 x 28 x 128        |
| CBAM_6                  |         |                            | 2277                 | 28 x 28 x 128        | 28 x 28 x 128        |
| Add_2                   | 0       |                            |                      | 28 x 28 x 128        | 28 x 28 x 128        |
| separable_conv_7        | 256     | 3/1                        | 34,176               | 28 x 28 x 128        | 28 x 28 x 256        |
| CBAM_7                  |         |                            | 8949                 | 25 x 28 x 256        | 28 x 28 x 256        |
| separable_conv_8        | 256     | 3/2                        | 34,176               | 28 x 28 x 128        | 28 x 28 x 256        |
| CBAM_8                  |         |                            | 8949                 | 25 x 28 x 256        | 28 x 28 x 256        |
| separable_conv_9        | 256     | 3/3                        | 34,176               | 28 x 28 x 128        | 28 x 28 x 256        |
| CBAM_9                  |         |                            | 8949                 | 25 x 28 x 256        | 28 x 28 x 256        |
| Add_3                   | 0       |                            |                      | 28 x 28 x 256        | 28 x 28 x 256        |
| separable_conv_10       | 256     | 3/1                        | 68,096               | 28 x 28 x 256        | 28 x 28 x 256        |
| separable_conv_11       | 256     | 3/1                        | 68,096               | 28 x 28 x 256        | 28 x 28 x 256        |
| Flatten                 | 0       |                            |                      | 28 x 28 x 256        | 200,704 x 1          |
| dense_1(1000)           |         |                            |                      | 200,705,000          | 1000 x 1             |
| dense_2 (256)           |         |                            |                      | 256,256              | 256 x 1              |

Figure 8. The model structure of ADSD-gcForest.
4. Experiments

4.1. Introduction of Datasets

The datasets used in the experiment were the Case Western Reserve University bearing dataset and the Canadian University of Ottawa bearing dataset. Two different bearings are contained in the Western Reserve University bearing dataset: drive end bearing SKF6205 and fan end bearing SKF6203. The drive end bearing included the two different sampling frequencies of 12 KHZ and 48 KHZ, while the sampling frequency of the fan end bearing was only 12 KHZ. Ten types of states are contained in each bearing dataset, which are normal state, inner ring failure, outer ring failure, and rolling element failure. Each fault state contains three different fault levels, represented by a fault diameter. A total of four different load conditions were applied when measuring the bearing data. A total of 8 normal samples, 53 outer ring damage samples, 23 inner ring damage samples and 11 rolling element damage samples were obtained. The Canadian Ottawa dataset is the bearing vibration signal of different health conditions measured under time-varying speed conditions, which had 36 datasets. The bearing conditions include: normal, inner ring failure and outer ring failure. The working speed conditions are speed increase, speed deceleration, deceleration after speed increase and speed increase after deceleration. Each dataset contains two channels, and channel 1 represents the vibration data measured by the accelerometer, channel 2 signifies the speed data measured by the encoder, the sampling frequency is 200 KHZ and the sampling duration is 10 s.

The drive end and the fan end bearing data of Western Reserve University used in this paper are at the sampling frequency of 12 KHZ, and for part of the dataset in Channel 1 of the University of Ottawa in Canada, the sample data used were randomly selected from the dataset, where B, IR and OR indicate that the fault location is located in the rolling element, inner ring and outer ring of the bearing, respectively. Moreover, 007, 014 and 021, respectively, indicate that the fault diameter is 0.1778 mm, 0.3556 mm and 0.5334 mm, and the number at the end indicates the size of the load. For example, “−1” means that the load is 1 horsepower. A total of 1000 samples were sampled for each fault category, and the sample ratio of training set to test set was 7:3. The details are shown in Table 2.

Table 2. Sample distribution table.

| Bearing Number | Dataset 1 | Dataset 2 | Dataset 3 | Dataset 4 | Dataset 5 | Dataset 6 |
|----------------|-----------|-----------|-----------|-----------|-----------|-----------|
| SKF6205        | Normal-1  | Normal-2  | Normal-3  | Normal-1  | Normal-2  | Normal-3  |
|                | B007-1    | B007-3    | B007-3    | B007-2    | B007-1    | B007-2    |
|                | B014-1    | B014-3    | B014-3    | B014-2    | B014-3    | B014-3    |
|                | B021-1    | B021-3    | B021-2    | B021-3    | B021-2    | B021-1    |
|                | IR007-1   | IR007-1   | IR007-2   | IR007-3   | IR007-2   | IR007-1   |
|                | IR014-2   | IR014-2   | IR014-2   | IR014-3   | IR014-3   | IR014-2   |
|                | IR021-3   | IR021-2   | IR021-2   | IR021-2   | IR021-3   | IR021-3   |
|                | OR007-1   | OR007-2   | OR007-1   | OR007-3   | OR007-3   | OR007-1   |
|                | OR014-2   | OR014-2   | OR014-1   | OR014-1   | OR014-2   | OR007-2   |
|                | OR021-3   | OR021-2   | OR021-2   | OR021-1   | OR021-2   | OR021-3   |
| SKF6203        | Normal-2  | Normal-2  | Normal-1  | Normal-3  | Normal-1  | Normal-2  |
|                | B007-1    | B007-1    | B007-3    | B007-2    | B007-1    | B007-3    |
|                | B014-3    | B014-2    | B014-2    | B014-3    | B014-2    | B014-3    |
|                | B021-3    | B021-1    | B021-2    | B021-3    | B021-2    | B021-1    |
|                | IR007-3   | IR007-2   | IR007-2   | IR007-3   | IR007-2   | IR007-1   |
|                | IR014-3   | IR014-1   | IR014-3   | IR014-1   | IR014-1   | IR014-1   |
|                | IR021-3   | IR021-3   | IR021-1   | IR021-1   | IR021-3   | IR021-3   |
|                | OR007-2   | OR007-2   | OR007-1   | OR007-1   | OR007-1   | OR007-2   |
|                | OR014-2   | OR014-2   | OR014-3   | OR014-2   | OR014-3   | OR007-1   |
|                | OR021-3   | OR021-3   | OR021-1   | OR021-2   | OR021-3   | OR021-1   |
Six noises of different intensities were added to the sample dataset, namely, Noise 1, Noise 2, Noise 3, Noise 4, Noise 5 and Noise 6. Each type of noise contains three different types of noise. The proportions of the three noises in the six noises were Gaussian noise with signal-to-noise ratios of $-4$, $-2$, $0$, $2$, $4$ and $6$, salt and pepper noise, with ratios of $0.3$, $0.25$, $0.2$, $0.15$, $0.1$ and $0.05$, and Cauchy noise with position parameter 0 and scale parameter 1. gcForest was set as the classifier in all comparison methods, and SigDSD-gcforest means that the Sigmoid function is the activation function of the convolutional layers. Similarly, the activation functions of the convolutional layers in ReluDSD-gcforest and PReluDSD-gcforest are Relu and PRelu, respectively. The parameter settings of the ADSD-gcForest model were as follows: the network training parameters were set at a learning rate of 0.00005, the number of batch processing was 380, the number of iterations was 350, Adam was used as the optimization algorithm, the sliding window dimension used in MGS was 240, the number of trees in the random forests of MGS was 35 and the number of trees in a single random forest in the cascade forest was 150. The diagnostic effect is analyzed by comparing the accuracy rate, F1 value and Area Under Curve (AUC) value of different diagnostic models after training. The accuracy rate is generally expressed as $\frac{TP + TN}{TP + TN + FP + FN}$, and F1 value is calculated as $\frac{2 \cdot FP}{2 \cdot TP + FP + FN}$, where TP refers to True Positives, FP represents True Negatives, FN indicates False Negatives and FP signifies False Positives. AUC is defined as the area under the area under curve. Generally, the higher the AUC value, the better the classification effect of the model.

4.2. Case Study 1: Performance of Drive End Bearing Fault Diagnosis

It can be seen from Figures 9 and 10 that when the noise environment is Noise 1, after the sample is trained by the ADSD-gcForest model, there are three categories of samples with low recognition rates, and there is also a small amount of aliasing in the T-SNE image. In other noise environments, there are only one or two fault categories with a low recognition rate. It can be seen from Table 3 that, compared to other methods, the ADSD-gcForest model achieves the highest fault accuracy rate and F1 value under various noises and different working conditions. Among them, the VGG16-gcForest model obtained the lowest accuracy and the F1 values, which is about 26–35% lower than those of the ADSD-gcForest model, while the accuracy and F1 values of the Res50-gcForest model are about 18% higher than the VGG16-gcForest model. Since the Relu function can better solve the network convergence problem than the Sigmoid function, the accuracy and F1 values of the Relu-gcForest model are about 0.6–0.7% higher than that of the SigDSD-gcForest model, and PRelu updates the weight according to the input data, which makes the network have a certain adaptive optimization capability. The accuracy and F1 values obtained after training is about 1.3% higher than that of the Relu-gcForest model, but its values are still lower than the ADSD-gcForest model. Figure 11 mainly describes the comparison of the AUC values of different methods. From Figure 11, it can be found that the AUC values of ADSD-gcForest under different noises are the highest and all are above 92%, indicating that the ADSD-gcForest model has a good fault diagnosis effect. It can be seen from the experimental results presented above that the ADSD-gcForest model can more accurately diagnose drive-end bearing failures under different working conditions and strong noise interference with a high accuracy rate.
Figure 9. Confusion matrix obtained by ADSD-gcforest model training (drive end). (a) Noise 1, (b) Noise 2, (c) Noise 3, (d) Noise 4, (e) Noise 5. (f) Noise 6.

Figure 10. T-SNE images obtained by ADSD-gcforest model training (drive end). (a) Noise 1, (b) Noise 2, (c) Noise 3, (d) Noise 4, (e) Noise 5. (f) Noise 6.
Table 3. Comparison table of the accuracy (AC) and F1 values of driver end data.

| Dataset   | Methods          | Dataset 2 N1(A/C/F1) | N2(A/C/F1) | N3(A/C/F1) | N4(A/C/F1) | N5(A/C/F1) | N6(A/C/F1) |
|-----------|------------------|-----------------------|------------|------------|------------|------------|------------|
| VGG16-gcForest | 65.10%/65.20% | 69.13%/68.78% | 71.20%/72.02% | 72.41%/73.10% | 75.36%/74.48% | 77.54%/76.98% |
| Res50-gcForest | 83.25%/82.66% | 86.42%/85.96% | 87.85%/88.12% | 89.22%/90.03% | 92.02%/92.05% | 93.33%/94.40% |
| SigDSD-gcForest | 92.43%/91.73% | 94.17%/93.92% | 95.95%/94.65% | 96.04%/96.11% | 96.35%/96.30% | 96.85%/96.80% |
| ReluDSD-gcForest | 92.85%/93.05% | 94.22%/93.88% | 96.20%/96.22% | 96.47%/96.50% | 96.70%/97.63% | 97.03%/96.98% |
| PReluDSD-gcForest | 93.09%/92.86% | 94.72%/94.70% | 96.62%/96.65% | 96.73%/96.80% | 96.83%/96.96% | 97.45%/97.52% |
| ASDS-gcForest | 94.32%/94.30% | 95.85%/95.78% | 97.70%/97.73% | 97.83%/98.03% | 97.92%/98.15% | 98.23%/98.28% |

Figure 11. Comparison figures of the AUC of the driver end data under Noise 1, 2, 3, 4, 5 and 6.
4.3. Case Study 2: Performance of Fan End Bearing Fault Diagnosis

It can be seen from Figures 12 and 13 that only when the noise environment is Noise 1, a few fault categories cannot be effectively identified. In other noise environments, the entire fault category can be accurately identified. It can be seen in Table 4 that the accuracy and F1 values obtained from the training of the VGG16-gcForest and Res50-gcForest models have dropped by approximately 1.5–1.6% compared to the driving end values. The overall accuracy and F1 value of the VGG16-gcForest model are between 61–75%. The accuracy and F1 values of the training of the SigDSD-gcForest, ReluDSD-gcForest and PreluDSD-gcForest models has also decreased. Among them, the most obvious decrease is SigDSD-gcForest, with a decrease from 0.3% to 0.4%, while the accuracy and F1 value of the PreluDSD-gcForest model drops by at least about 1.3–1.5%. The accuracy value and F1 values of the ADSD-gcForest model are the highest, and these values are similar to case study 1. Figure 14 depicts the AUC values obtained by different diagnostic methods under different noises. It can be found that the AUC values obtained by the ADSD-gcForest model are still the highest, which are close to those obtained in case study 1. Through the experimental results presented above, it can be found that the ADSD-gcForest model proposed in this paper can basically realize an effective fault diagnosis for different bearings under multiple working conditions.

![Confusion matrix obtained by ADSD-gcForest model training (fan end).](image)

Figure 12. Confusion matrix obtained by ADSD-gcForest model training (fan end). (a) Noise 1, (b) Noise 2, (c) Noise 3, (d) Noise 4, (e) Noise 5, (f) Noise 6.
Figure 13. T-SNE images obtained by ADSD-gcForest model training (fan end). (a) Noise 1, (b) Noise 2, (c) Noise 3, (d) Noise 4, (e) Noise 5. (f) Noise 6.

Figure 14. Comparison figures of the AUC of the fan end data under Noise 1, 2, 3, 4, 5 and 6.
Table 4. Comparison table of the accuracy (AC) and F1 values of the fan end data.

| Dataset | Methods          | N1(AC/F1) | N2(AC/F1) | N3(AC/F1) | N4(AC/F1) | N5(AC/F1) | N6(AC/F1) |
|---------|------------------|-----------|-----------|-----------|-----------|-----------|-----------|
|         | VGG16-gcForest   | 61.87%/61.80% | 64.84%/64.75% | 67.45%/68.12% | 69.85%/70.23% | 73.46%/73.32% | 75.75%/75.82% |
|         | Res50-gcForest   | 79.75%/80.04% | 81.28%/81.42% | 84.86%/84.90% | 87.91%/88.34% | 91.03%/91.06% | 92.46%/92.50% |
|         | SigDSD-gcForest  | 91.52%/91.68% | 92.53%/93.46% | 94.48%/95.18% | 95.56%/95.63% | 96.01%/96.20% | 96.65%/96.60% |
|         | ReluDSD-gcForest | 92.78%/92.87% | 93.71%/93.72% | 95.49%/95.64% | 96.52%/96.60% | 97.04%/97.06% | 97.75%/97.80% |
|         | PReluDSD-gcForest| 93.24%/93.29% | 94.78%/95.10% | 96.25%/96.68% | 97.23%/97.20% | 97.72%/97.80% | 98.34%/98.25% |
|         | ADSD-gcForest    | 93.85%/93.92% | 95.32%/95.54% | 96.81%/96.98% | 97.26%/97.39% | 97.83%/98.00% | 98.44%/98.40% |

4.4. Case Study 3: Performance of the Ottawa Bearing Dataset

In order to further test the generalization and robustness of the ADSD-gcForest model, case study 3 focused on the University of Ottawa dataset, which was specifically divided into six datasets. The setting method of adding noise was the same as case study 1. The specific sample types are shown in Table 5. There were three operation conditions of the bearings in the datasets, i.e., normal (H), inner race fault (I) and out race fault (O), and also contained four speed transformation conditions, i.e., speed up (A), slow down (B), speed up and slow down (C) and slow down and speed up (D). The noise setting used in case study 3 is the same as the case study 1. The training parameter settings of the ADSD-gcForest model are as follows: the network training parameters were set to a learning rate of 0.00005, the number of batch processing was 550, the number of iterations was 350, Adam was used as the optimization algorithm, the sliding window dimension used in MGS was 240, the number of trees in the MGS random forest was 35 and the number of trees in a single random forest in the cascade forest was 150.
Table 5. Sample distribution table.

| The Name of Dataset | Dataset 1 | Dataset 2 | Dataset 3 | Dataset 4 | Dataset 5 | Dataset 6 |
|---------------------|-----------|-----------|-----------|-----------|-----------|-----------|
| University of Ottawa bearing data | HD-1      | HA-1      | HB-1      | HA-1      | HC-1      | HD-1      |
|                     | HA-1      | HB-1      | HC-1      | HB-1      | HA-1      | HB-1      |
|                     | HB-1      | HD-1      | HA-1      | HD-1      | HB-1      | HC-1      |
|                     | IC-1      | IA-1      | IC-1      | IB-1      | IB-1      | ID-1      |
|                     | ID-1      | IB-1      | IA-1      | IC-1      | IA-1      | IA-1      |
|                     | IB-1      | ID-1      | ID-1      | IA-1      | IB-1      | IB-1      |
|                     | OB-1      | OD-1      | OB-1      | OB-1      | OA-1      | OD-1      |
|                     | OD-1      | OC-1      | OA-1      | OD-1      | OC-1      | OB-1      |
|                     | OA-1      | OB-1      | OD-1      | OC-1      | OD-1      | OC-1      |
|                     | HC-1      | IA-1      | HA-1      | IB-1      | OD-1      | OB-1      |

It can be seen from Figure 15 that, compared to case study 1 and case study 2, when the noise environment is Noise 1 and Noise 2, the degree of discrimination of some fault categories is lower, but in other noise environments, the fault categories can be accurately classified. It can be seen from Figures 16 and 17 that the training accuracy of the ADSD-gcForest model is the highest and the value is relatively stable, while the fluctuation is small, which is consistent with the values in Table 6. At the same time, it can be found from Table 6 that the accuracy and F1 values obtained by training the VGG16-gcForest and Res50-gcForest models are significantly lower than case study 1 and case study 2. In Figures 16 and 17, the accuracy of the two models also fluctuates significantly, and the accuracy of the other three models is more accurate. The rate values have also decreased, but the value fluctuations are relatively small. Figure 18 reflects the AUC values of different diagnostic models, from which it can be found that the AUC values of the ADSD-gcForest model are basically similar to the first two cases, but other diagnostic models have decreased. Through comparative experiments of three groups of different bearings, it can be seen that under different noise conditions and for bearing data under different working conditions, in one way, the ADSD-gcForest model can achieve effective fault feature extraction, while in another way, the use of the Meta-ACON activation function can easily and efficiently complete the self-adaptive optimization of the model structure and realize more accurate fault diagnosis.

Figure 15. T-SNE images obtained by ADSD-gcforest model training (Ottawa Bearing). (a) Noise 1, (b) Noise 2, (c) Noise 3, (d) Noise 4, (e) Noise 5, (f) Noise 6.
Figure 16. Box plots of accuracy values under different noise conditions. (a) Noise 1, (b) Noise 2.

Figure 17. Box plots of accuracy values under different noise conditions. (a) Noise 3, (b) Noise 4, (c) Noise 5, (d) Noise 6.
Table 6. Comparison table of the accuracy (AC) and F1 values of the Ottawa dataset.

| Dataset | Methods | Dataset 1 | Dataset 2 | Dataset 3 | Dataset 4 | Dataset 5 | Dataset 6 |
|---------|---------|----------|----------|----------|----------|----------|----------|
|         | N1(AC/F1) | N2(AC/F1) | N3(AC/F1) | N4(AC/F1) | N5(AC/F1) | N6(AC/F1) |
| VGG16-gcForest | 56.23%/56.20% | 60.39%/60.34% | 63.36%/63.72% | 65.63%/65.59% | 68.42%/68.45% | 70.28%/70.36% |
| Res50-gcForest | 78.29%/78.26% | 82.53%/82.57% | 84.31%/84.59% | 85.42%/85.31% | 87.12%/87.18% | 88.03%/87.93% |
| SigDS-gcForest | 91.23%/91.36% | 92.39%/92.58% | 94.52%/95.67% | 95.26%/95.68% | 96.01%/96.19% | 96.75%/97.00% |
| ReluDS-gcForest | 91.79%/91.66% | 93.42%/93.35% | 95.01%/95.16% | 95.76%/95.72% | 96.42%/96.48% | 97.03%/97.18% |
| PReluDS-gcForest | 92.34%/92.46% | 93.44%/93.67% | 95.47%/95.55% | 96.12%/96.39% | 97.03%/98.98% | 97.35%/97.36% |
| ADSD-gcForest | 93.42%/93.39 | 94.76%/94.58% | 96.08%/96.06% | 96.89%/96.92% | 97.62%/97.58% | 98.42%/98.48% |
| VGG16-gcForest | 56.80%/56.68% | 58.55%/59.04% | 62.53%/62.68% | 64.33%/65.22% | 66.86%/67.68% | 68.42%/68.40% |
| Res50-gcForest | 76.73%/76.55% | 78.42%/78.61% | 81.18%/81.26% | 83.92%/83.90% | 86.35%/86.42% | 88.26%/88.37% |
| SigDS-gcForest | 90.08%/88.89% | 90.91%/91.26% | 92.58%/92.87% | 94.27%/94.38% | 95.31%/95.36% | 96.26%/96.28% |
| ReluDS-gcForest | 90.02%/89.52% | 91.54%/91.69% | 93.18%/92.89% | 94.82%/94.80% | 96.05%/96.18% | 96.86%/97.67% |
| PReluDS-gcForest | 90.16%/91.02% | 92.32%/92.21% | 94.84%/94.56% | 95.47%/95.32% | 96.76%/97.91% | 97.32%/97.42% |
| ADSD-gcForest | 91.85%/91.29% | 93.49%/93.74% | 95.68%/95.88% | 96.31%/96.28% | 97.45%/97.52% | 98.23%/98.38% |
| VGG16-gcForest | 57.34%/57.49% | 59.56%/59.63% | 61.16%/61.19% | 63.47%/63.30% | 65.32%/65.71% | 67.92%/67.94% |
| Res50-gcForest | 75.26%/75.36% | 77.56%/77.49% | 80.64%/80.79% | 83.94%/84.06% | 85.16%/85.67% | 87.85%/88.06% |
| SigDS-gcForest | 91.53%/91.68% | 92.20%/92.34% | 93.23%/94.36% | 94.78%/94.82% | 95.63%/95.56% | 96.28%/96.34% |
| ReluDS-gcForest | 91.06%/91.64% | 92.76%/92.84% | 93.74%/93.70% | 95.58%/95.88% | 96.77%/97.24% | 97.96%/98.38% |
| PReluDS-gcForest | 91.60%/91.65% | 93.26%/93.36% | 94.43%/94.58% | 96.03%/96.15% | 97.66%/97.72% | 99.64%/99.67% |
| ADSD-gcForest | 93.42%/93.62% | 94.60%/94.26% | 95.81%/95.88% | 96.63%/96.58% | 97.80%/97.84% | 98.32%/98.30% |
| VGG16-gcForest | 58.65%/58.79% | 58.52%/58.16% | 61.83%/62.05% | 62.55%/63.18% | 64.67%/65.25% | 66.86%/66.79% |
| Res50-gcForest | 74.00%/74.59% | 76.82%/75.99% | 82.35%/83.59% | 84.54%/85.69% | 85.46%/85.96% | 89.14%/89.19% |
| SigDS-gcForest | 91.01%/91.08% | 93.06%/93.28% | 95.15%/95.28% | 95.70%/95.76% | 96.02%/96.34% | 96.55%/96.64% |
| ReluDS-gcForest | 91.57%/91.68% | 93.62%/94.68% | 95.65%/95.78% | 96.04%/96.29% | 96.66%/96.64% | 97.05%/97.14% |
| PReluDS-gcForest | 92.02%/91.98% | 94.22%/94.26% | 96.03%/96.38% | 96.48%/96.56% | 97.21%/97.36% | 97.43%/97.58% |
| ADSD-gcForest | 92.52%/93.76% | 94.76%/94.88% | 96.56%/96.49% | 97.03%/96.69% | 97.85%/97.68% | 98.42%/98.64% |

Figure 18. Comparison figures of the AUC of the Ottawa dataset under Noise 1, 2, 3, 4, 5 and 6.
5. Conclusions

This paper proposes an adaptive ADSD-gcForest model. The model uses the VGG network as the basic framework. Multi-scale features of input samples can be extracted through deep separable dilated convolution, and then the CBAM to focus the core features is combined at different scales, the Meta-ACON activation function is integrated into all convolution layers in the network, so that the model can be optimized adaptively according to different input data, and the gcForest as the final classifier can provide the final result. In the experimental part of this paper, datasets of Western Reserve University and University of Ottawa are used, including three bearing data, and it can be seen that faults of different types of bearings under strong noise and multiple load conditions can be effectively diagnosed by the ADSD-gcForest model. This shows that the model proposed in this paper has good robustness. It can also be found that the method proposed in this paper has better improved the migration ability of the model, simplified the design process of the diagnostic model and effectively avoided the problem of repeatedly modifying the model structure.

In modern industrial production, multiple bearings are often required to work together; thus, the effective fault diagnosis of multiple bearings is a hot research topic. The ADSD-gcForest model proposed in this paper can simply optimize the model structure according to different bearing data with the help of the Meta-ACON activation function. It has a certain industrial application value, but the addition of the Meta-ACON activation function also increases the number of parameters of the model, which leads to a longer training time. Therefore, how to reduce the training parameters of the Meta-ACON activation function under the premise of ensuring high accuracy will become the focus of future research.

Author Contributions: Conceptualization: S.Z.; Methodology: S.Z. and Z.W.; Formal analysis and investigation: S.Z.; Writing—original draft preparation: S.Z.; Writing—review and editing: D.G. and S.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The used data of bearing fault can be found in CWRU Data Center and University of Ottawa datasets are available online: https://csegroups.case.edu/bearingdatacenter/pages/welcome-case-western-reserve-university-bearing-data-center-website (accessed 18 October 2021), https://data.mendeley.com/datasets/v43hmbwxpm/1 (accessed 20 October 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Fan, J.; Qi, Y.; Liu, L.; Gao, X.; Li, Y. Application of an information fusion scheme for rolling element bearing fault diagnosis. *Meas. Sci. Technol.* 2021, 32, 075013. [CrossRef]
2. Niu, G.; Wang, X.; Golda, M.; Mastro, S.; Zhang, B. An optimized adaptive PReLU-DBN for rolling element bearing fault diagnosis. *Neurocomputing* 2021, 445, 26–34. [CrossRef]
3. Jie, D.; Zheng, G.; Zhang, Y.; Ding, X.; Wang, L. Spectral kurtosis based on evolutionary digital filter in the application of rolling element bearing fault diagnosis. *Int. J. Hydromechatronics* 2021, 4, 27–42. [CrossRef]
4. Zhao, X.; Qin, Y.; Fu, H.; Jia, L.; Zhang, X. Blind source extraction based on EMD and temporal correlation for rolling element bearing fault diagnosis. *Smart Resilient Transp.* 2021, 3, 52–65. [CrossRef]
5. Hou, W.; Ye, M.; Li, W. Rolling bearing fault classification based on improved stack noise reduction self-encoding. *Chin. J. Mech. Eng.* 2018, 54, 87–96. [CrossRef]
6. Shao, H.; Jiang, H.; Zhang, X.; Niu, M. Rolling bearing fault diagnosis using an optimization deep belief network. *Meas. Sci. Technol.* 2015, 26, 115002. [CrossRef]
7. Liang, T.; Wu, S.; Duan, W.; Zhang, R. Bearing fault diagnosis based on improved ensemble learning and deep belief network. *J. Phys. Conf. Ser.* 2018, 1074, 012154. [CrossRef]
8. Ma, M.; Chen, X.; Wang, S.; Liu, Y.; Li, W. Bearing degradation assessment based on weibull distribution and deep belief network. In Proceedings of the IEEE International Symposium on Flexible Automation, Cleveland, OH, USA, 1–3 August 2016; pp. 382–385.
9. Shao, S.; Sun, W.; Wang, P.; Gao, R.X.; Yan, R. Learning features from vibration signals for induction motor fault diagnosis. In Proceedings of the IEEE International Symposium on Flexible Automation, Cleveland, OH, USA, 1–3 August 2016; pp. 71–76.
10. Han, T.; Yuan, J.H.; Tang, J.; An, L.Z. Intelligent composite fault diagnosis method of rolling bearing based on MWT and CNN. Mech. Transm. 2016, 40, 139–143.
11. Liang, M.; Cao, P.; Tang, J. Tang. Rolling bearing fault diagnosis based on feature fusion with parallel convolutional neural network. Int. J. Adv. Manuf. Technol. 2020, 112, 819–831. [CrossRef]
12. Pan, H.; He, X.; Tang, S.; Meng, F. An improved bearing fault diagnosis method using one-dimensional CNN and LSTM. J. Mech. Eng. 2018, 64, 443–452.
13. Zhang, L.; Jing, L.; Xu, W.; Tan, J. Rolling bearing fault diagnosis based on convolutional noise reduction autoencoder and CNN. Modul. Mach. Tool Autom. Manuf. Technol. 2019, 6, 58–62.
14. Yu, L.; Qu, J.; Gao, F.; Tian, Y. A novel hierarchical algorithm for bearing fault diagnosis based on stacked LSTM. Shock. Vib. 2019, 2019, 2756284. [CrossRef] [PubMed]
15. Zhao, R.; Yan, R.; Wang, J.; Mao, K. Learning to monitor machine health with convolutional bi-directional lstm networks. Sensors 2017, 17, 273. [CrossRef] [PubMed]
16. Yan, X.; Xu, Y.; Jia, M. Intelligent Fault Diagnosis of Rolling-Element Bearings Using a Self-Adaptive Hierarchical Multiscale Fuzzy Entropy. Entropy 2021, 23, 1128. [CrossRef] [PubMed]
17. Yong, Z.; Zhang, X.; Da, N. Research on 3D Object Detection Method Based on Convolutional Attention Mechanism. J. Phys. Conf. Ser. 2021, 1848, 012097. [CrossRef]
18. Cao, Q.; Yu, L.; Wang, Z.; Zhan, S.; Quan, H.; Yu, Y.; Khan, Z.; Koubaa, A. Wild Animal Information Collection Based on Depthwise Separable Convolution in Software Defined IoT Networks. Electronics 2021, 10, 2091. [CrossRef]
19. Ma, N.; Zhang, X.; Liu, M.; Sun, J. Activate or Not: Learning Customized Activation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020.
20. Zhang, N.; Cui, F.; Jiang, B.; He, X. Rotating machinery gearbox fault diagnosis method integrating SDP and CNN. Ind. Control. Comput. 2021, 34, 89–91.
21. Zhao, L.; Xu, L.; Liu, J.; Liu, J.; Huang, X. Transformer mechanical fault diagnosis method based on point symmetry transformation and image matching. Trans. Chin. Soc. Electr. Eng. 2021, 36, 3614–3626.
22. Yin, Q.; Yang, W.; Ran, M.; Wang, S. FD-SSD: An improved SSD object detection algorithm based on feature fusion and dilated convolution. Signal Processing Image Commun. 2021, 98, 116402. [CrossRef]
23. Yuhui, Z.; Mengyao, C.; Yuelin, C.; Zhaoqian, L.; Yao, L.; Kedi, L. An Automatic Recognition Method of Fruits and Vegetables Based on Depthwise Separable Convolution Neural Network. J. Phys. Conf. Ser. 2021, 1871, 012075. [CrossRef]
24. Feng, Y.; Gao, P. Generative Robotic Grasping Using Depthwise Separable Convolution. Comput. Electr. Eng. 2021, 94, 107318. [CrossRef]
25. Liu, T.; Pang, B.; Zhang, L.; Yang, W.; Sun, X. Sea Surface Object Detection Algorithm Based on YOLO v4 Fused with Reverse Depthwise Separable Convolution (RDSC) for USV. J. Mar. Sci. Eng. 2021, 9, 753. [CrossRef]
26. Chen, Y.; Zhang, X.; Chen, W.; Li, Y.; Wang, J. Research on Recognition of Fly Species Based on Improved RetinaNet and CBAM. IEEE Access 2020, 8, 102907–102919. [CrossRef]
27. Canayaz, M. C+EffxNet: A novel hybrid approach for COVID-19 diagnosis on CT images based on CBAM and EfficientNet. Chaos Solitons Fractals 2021, 151, 111310. [CrossRef] [PubMed]
28. Niu, C.; Nan, F.; Wang, X. A super resolution frontal face generation model based on 3DDFA and CBAM. Display 2021, 69, 102043. [CrossRef]
29. Sun, Z.; Li, M.; Zhang, J.; Hu, B.; Qi, G.; Zhu, Y. Transient Voltage Stability Assessment Method based on gcForest. J. Phys. Conf. Ser. 2021, 1914, 012025. [CrossRef]
30. Liu, H.; Zhang, N.; Jia, S.; Xu, D.; Gao, W. Small sample color fundus image quality assessment based on gcForest. Multimed. Tools Appl. 2020, 80, 17441–17459. [CrossRef]