Article

Research on the Service Condition Monitoring Method of Rolling Bearings Based on Isomorphic Data Fusion

Yanfei Zhang [1,2,*], Yang Liu [1], Mingqi Yang [2], Xiaoyang Feng [3], Qianxiang Zhu [4] and Lingfei Kong [1]

[1] School of Mechanical and Precision Instrument Engineering, Xi'an University of Technology, Xi'an 710048, China; 2210221250@stu.xaut.edu.cn (Y.L.); lingfeikong@xaut.edu.cn (L.K.)
[2] Luoyang Bearing Science & Technology Co., Ltd., Luoyang 471039, China; 15937904981@163.com
[3] Shaanxi Robot Automation Technology Co., Ltd., Xi'an 710061, China; fxy@xknc.net
[4] Xi'an Research Institute Co., Ltd., China Coal Technology and Engineering Group Corp, Xi'an 710048, China; zhuqianxiang@cctegxian.com
* Correspondence: yfzhang@xaut.edu.cn

Abstract: In order to solve the problem that it is difficult for a single sensor to accurately characterize the running state of rotating bearings under complex working conditions, this paper proposes a data-level fusion method based on multi-source isomorphic sensors to monitor spindle bearings. First, new vibration signals in the X,Y,Z direction were obtained through the process of decomposing, de-noise, and reconstructing. Second, the PCA algorithm was used to select the time-domain and frequency-domain features of the vibration signals, construct the feature matrix, and perform dimensionality reduction in the feature matrix. Finally, the entropy weight method was introduced to obtain the initial weights of the three directions as the inputs of the adaptive function. The chaotic particle swarm optimization algorithm proposed in this paper helps particles jump out of the local optimum. Chaotic mapping is used to initialize the velocity and position of the particles, which calculates globally optimal weights in three directions. In order to extract bearing signal features more accurately and efficiently, a DenseNet and Transformer (DAT) feature extraction model is proposed to deal with the complex changes and noise interference of bearing signals. Through the open data set of Jiangnan University and the data collected by our own experimental platform, the maximum accuracy of the DAT model was verified to be 100%.

Keywords: multi-source; entropy weighting; chaotic particle swarm optimization algorithm; data-level fusion; feature extraction

1. Introduction

Machine tools are important and necessary instruments in the equipment manufacturing sector. The spindle, which is the machine tool's core component, determines its precision and productivity. Bearing assembly precision and performance are critical because they determine the spindle's running condition and performance, which influences the machine tool's overall machining quality and effectiveness. [1–3]. To forecast the dynamic performance of the bearing-rotor system, Ma S et al. [4] developed a dynamic model based on SFBE. By describing SFBE-specific physical properties, this model provides real-time coupling and the synchronous solution of bearing and rotor models. Fang B et al. [5] proposed a generalized mathematical model of DR-ACBB under three different configurations to study the variation rule of nonlinear stiffness. It resulted in the skewed running of bearings in the service, which can very easily cause bearing failures. This is due to long-term service in harsh environments such as variable loads, high temperatures, and impacts, as well as under the influence of factors like manufacturing errors, assembly accuracy, and human operation errors. Relying on mathematical models alone is limited and no longer allows for the complete condition characterization of bearings. To ensure the safe operation of machine tools and to boost production, the reliable and effective real-time condition

monitoring of bearings is crucial [6]. Machine learning has developed into a very powerful classification tool with the advancement of computer technology. Machine learning can delve deeper into potentially advantageous information in data since computers can handle enormous amounts of data [7,8]. Traditional shallow machine-learning techniques do have some limitations. These techniques usually require a lot of prior knowledge, which makes selecting and extracting features challenging [9,10].

End-to-end deep learning methodologies have been increasingly brought into the field of fault diagnosis in recent years as a result of the impact of the artificial intelligence (AI) wave. Deep learning techniques, as opposed to conventional approaches, offer fresh perspectives and opportunities for defect detection research by automatically learning feature representations and eliminating inevitable ambiguities in the manual feature extraction process [11]. One of these representative deep learning methods, the convolutional neural network (CNN), is a subclass of the feed-forward neural network that includes convolutional computation and has a deep structure [12]. This algorithm is capable of representational learning and can categorize input data according to its hierarchical structure in a translation-invariant manner. Janssens [13] suggested a three-layer CNN model for bearing defect identification based on vibration signals. Prior to training the model and feeding data into the network model, these data were discretely Fourier-converted. Gu [14] suggested that the 1-DCNN and LSTM two channels be fed raw vibration signals to fuse feature information in both the temporal and spatial dimensions, thereby classifying the bearing problems. Zhang et al. [15] proposed a method to monitor the uneven operating conditions of bearings based on a two-channel fusion of the improved DenseNet network, which realizes the fusion of features in the frequency domain and the time-frequency domain. This method addresses the issue that traditional bearing fault diagnosis methods are insufficient to extract key features under strong noise and variable loads. Aiming at the above references, the feature extraction model in this article introduces DenseNet and Transformer modules to improve this model's ability to deal with complex working conditions.

The multi-sensor measurement and sensing system is a complex information processing system that integrates target measurements, data processing, and information fusion. It is widely used in the fields of industrial system monitoring [16], fault diagnosis [17], spatial localization [18], and environmental observation [19]. Among them, at the data level, the fusion of homologous and homomorphic multi-sensor sensing sequences is one of the key elements of this system. By fusing data from multiple sensors, the accuracy and reliability of information can be improved, and then more accurate measurement and sensing results can be realized, which provides important support for achieving efficient data processing and decision making [20]. In the field of intelligent manufacturing, data fusion technology effectively improves people's processing ability and utilization efficiency of industrial big data, where multi-source data have the characteristics of comprehensively describing the target alongside complementary data, and its fusion operation can improve the decision-making credibility and anti-interference ability of the model, reduce the redundancy existing in multi-source data, and reduce the waste of storage resources [21].

Data fusion methods of multi-source homogeneous sensors have been introduced to comprehensively characterize the operating state of rolling bearings because it is challenging for a single sensor to accurately characterize the operating state of machine tool spindle bearings under complex working conditions. Common fusion methods for homogeneous- and homogeneous-type multi-sensor sensing sequences include the weighted averaging method [22,23], Bayesian estimation [24], maximum likelihood estimation [25], Kalman filtering [26], neural network [27] and fuzzy logic [28]. Among these, the weighted average method is suitable for data layer fusion, but the distribution of weights has a significant impact on the fusion effect [22]. Bayesian estimation, maximum likelihood estimation, and other statistical methods require a priori knowledge of the target object. Kalman filtering requires that the system model and statistical characteristics of noise are known, and it cannot deal with the problem of adding new sensors. Neural network-based methods require training and learning, and their applicability is affected by the number of input

dimensions and the number of neurons and cannot handle input source variations. Fuzzy C-mean clustering-based methods are computationally straightforward [28], do not require a priori knowledge and model limitations, and can be applied online, but their results depend on the precise estimation of the number of clusters. Neural network-based methods require training and learning, and their applicability is affected by the number of input dimensions and neurons and cannot handle input source variations.

Su [29] proposed a homogeneous multi-sensor online fusion method based on improved fuzzy clustering and the aforementioned analysis. This method uses a robust fuzzy clustering method that introduces noise classes to analyze multiple sources of data simultaneously and does not depend on the number of clusters set in the traditional fuzzy clustering fusion method. A multi-sensor data fusion approach based on an adaptive weighting algorithm was proposed by Tang [30]. This method fuses signals from several sensors using an adaptive weighting algorithm and then uses a Kalman filtering algorithm to decrease noise in the output. In their algorithm, Cai et al. [31] combined measurement data preprocessing with improved batch estimation adaptive and weighted data fusion, introducing environmental factors and enhancing the batch estimation algorithm to determine the ideal monitoring value of individual sensors, and realizing adaptive weighted data fusion in accordance with the principle of optimal weight allocation. Zhu et al. [32] proposed a multi-sensor data fusion algorithm based on wavelet noise reduction and adaptive weighting to address the issues of large error, conflict, and redundancy in the multi-node data acquisition of greenhouse environmental information. The proposed algorithm processed the collected data through wavelet noise reduction to make it have good smoothness and stability and fused multi-sensor data using the adaptive weighting algorithm. With the guidance of the above literature, this article constructs a fusion algorithm based on weighted fusion, which is based on the entropy weighting method and chaotic particle swarm search.

In conclusion, this paper proposes a data-level fusion method based on multi-source isomorphic sensors to monitor the running state of rolling bearings and constructs a DAT feature extraction model for the deep feature extraction of fused data to detect this bearing's service state. It is challenging for a single sensor to accurately characterize the service state of machine tool spindle bearings under complex working conditions. In order to improve the accuracy, reliability, coverage, time-domain continuity, and consistency of data, as well as the fault tolerance and robustness of the system, a data-level fusion method with multi-source isomorphic sensors is proposed to monitor the operational status of rolling bearings. We created the DAT deep learning model (DenseNet and Transformer, DAT), which introduces the serial combination of DenseNet and Transformer modules to enable feature reuse and improve the model's capacity for handling time-series data, enabling more sophisticated feature extraction and transformation.

2. Relevant Theoretical Approaches

2.1. Wavelet Packet Denoising

Wavelet packet decomposition is substantially more effective than wavelet analysis's capacity to analyze signals since it decomposes both the high- and low-frequency portions of the signal. As seen in Figure 1, the following is an illustration of a four-layer wavelet packet decomposition, with a denoting the low-frequency portion and b denoting the high-frequency portion [33].
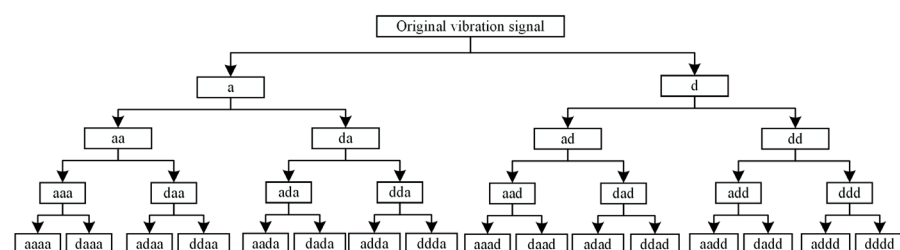

Figure 1. Four-layer wavelet packet decomposition.

In wavelet packet decomposition, $\mu_0 = \varphi(t)$, $\mu_1 = \psi(t)$ where $\varphi(t)$ and $\psi(t)$ denote the scale function and wavelet function, respectively, and $\{h_n\}_{n \in Z}$ and $\{g_n\}_{n \in Z}$ denotes the low-pass and high-pass filters, respectively; a set of functions known as wavelet packets can be defined by μ_0, μ_1, h, g at a fixed scale:

$$\mu_{2n}(t) = \sqrt{2}\sum_k h_k \mu_n(2t - k) \tag{1}$$

$$\mu_{2n+1}(t) = \sqrt{2}\sum_k g_k \mu_n(2t - k) \tag{2}$$

where $\mu_n, n = 0, 1, 2, \ldots n$. is called the wavelet packet and is determined by the orthogonal scale function $\mu_0 = \varphi(t)$.

A noisy signal is provided as follows:

$$s = x + n \tag{3}$$

where s is the measured noise signal, x is the original signal, n is the noise, and the essence of signal denoising is to estimate the original signal x based on the detected noise signals. The corresponding wavelet packet threshold denoising steps are as follows.

$$\begin{cases} y = W(s) \\ \widetilde{y} = D(y, \lambda) \\ \widetilde{x} = \overline{W}\left(\widetilde{y}\right) \end{cases} \tag{4}$$

where W and $\overline{W}$ denote the wavelet packet transform and its inverse, respectively, λ is the threshold, and D is the signal thresholding denoising process.

$$SNR = 10lg\left[\frac{\sum_n x^2(n)}{\sum_n (\hat{x}(n) - x(n))^2}\right] \tag{5}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_n [\hat{x}(n) - x(n)]^2} \tag{6}$$

In this formula, SNR represents the signal-to-noise ratio, which is the ratio of the energy of the useful signal to the energy of noise; the larger the SNR is, the smaller the noise mixed in the measured signal is. RMSE represents the root mean square error, which is the root mean square error between the signal after reconstruction and the original signal, and the smaller this value is, the better the de-noising effect is; $\hat{x}(n)$ is the signal with noise, and $x(n)$ is the original signal.

The original vibration signal can be decomposed to a maximum of 15 layers. After comparing the signal-to-noise ratio and the root mean square error of the various layers, the decomposition of the wavelet packet after four layers was selected. Too many layers result in the loss of actual useful information, and too few layers are not able to play a role in improving the signal-to-noise ratio.

2.2. Entropy Weighting Method

The entropy weight method is a multi-criteria decision analysis method that realizes the comprehensive evaluation of each indicator by calculating the entropy value and weight of the indicator. This method does not need to standardize the data and is suitable for situations where each indicator has a different scale and a different direction. The core idea of the entropy weighting method is that the smaller the entropy value of an indicator is, the more informative it is, and the greater the impact on the comprehensive evaluation it has; the weight of the indicator is calculated according to the proportion of the entropy value of each indicator [34], and the features extracted in this paper are shown in Table 1.

Table 1. Feature extraction.

Feature Classification	Feature Extraction
Time domain features	Maximum value, minimum value, peak-to-peak value, average value, absolute average value, root mean square, variance, standard deviation, steepness, skewness, peak factor, waveform factor, pulse factor, margin factor
Frequency domain features	Mean frequency, mean square frequency, root mean square frequency, frequency variance, frequency standard deviation

By extracting the time-domain and frequency-domain features of the original signal, the matrix can be defined $X = \{X_{ij}\}$, where i is the number of sensors, $i = 1, 2, \ldots, n$; j is the number of feature indicators, $j = 1, 2, \ldots, m$; and then the feature matrix can be expressed as:

$$X = \begin{bmatrix} X_{11} & X_{12} & \ldots & X_{1m} \\ X_{21} & X_{22} & \ldots & X_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \ldots & X_{nm} \end{bmatrix} \tag{7}$$

The weight of the j-th feature indicator under the i-th sensor is as follows:

$$P_{ij} = \frac{X_{ij}}{\sum_{i=1}^{N} x_{ij}} \tag{8}$$

The entropy value under the i-th sensor can be calculated as shown below:

$$H_i = -\frac{1}{\ln N}\sum_{i=1}^{N} P_{ij}\ln P_{ij} \tag{9}$$

where $P_{ij} \ln P_{ij}$ is considered to be 0 if $P_{ij} = 0$.

The weight of the i-th sensor can be calculated as follows:

$$W_i = \frac{1 - H_i}{n - \sum_{i=1}^{n} H_i} \tag{10}$$

2.3. Principle of the PCA Downscaling Algorithm

PCA (Principal Component Analysis) is a commonly used data dimensionality reduction method, which can reduce high-dimensional data into a low-dimensional space while trying to retain the information of the data [35]. The PCA dimensionality reduction algorithm flow of this paper is shown in Figure 2.
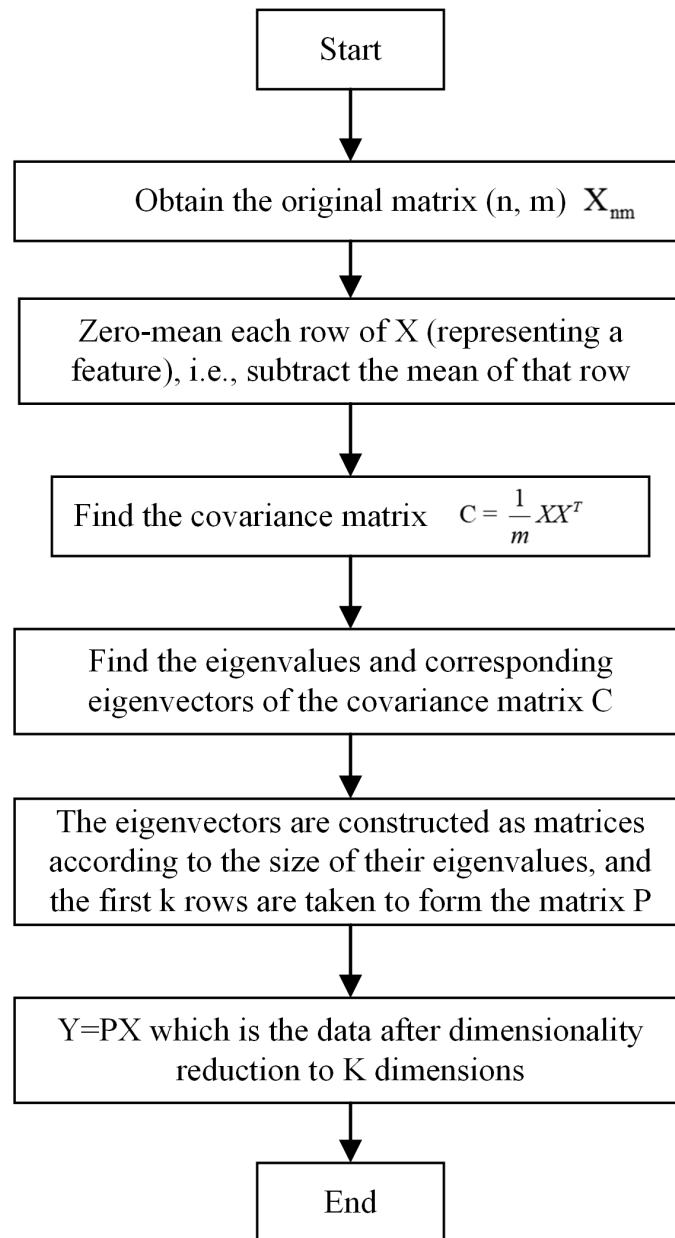
2.4. Chaos Mapping

The basic idea of the chaotic optimization algorithm is to map chaotic variables from a chaotic space to a solution space and then search for this using the characteristics of chaotic variables with traversability, randomness, and regularity. The chaotic optimization algorithm has the characteristics of not appearing sensitive to the initial value, it is easy to jump out of the local minima, and has a fast search speed, high computational accuracy, and global asymptotic convergence. Chaotic sequences commonly used in the field of group intelligence mainly include Logistic mapping, PWLCM mapping, Singer mapping, Sine mapping, etc., and in this method, Logistic mapping was chosen to optimize the particle swarm algorithm [36].

Logistic mapping is implemented as follows:

$$z_{k+1} = \mu z_k(1 - z_k) \tag{11}$$

where $z_0 \notin \{0, 0.25, 0.5, 0.75, 1.0\}, \mu \in [0, 4]$.

```
┌─────────────────────────┐
│          Start          │
└─────────────────────────┘
             │
             ▼
┌───────────────────────────────────────────┐
│   Obtain the original matrix (n, m)  Xₙₘ   │
└───────────────────────────────────────────┘
             │
             ▼
┌───────────────────────────────────────────┐
│  Zero-mean each row of X (representing a   │
│  feature), i.e., subtract the mean of that row │
└───────────────────────────────────────────┘
             │
             ▼
┌───────────────────────────────────────────┐
│  Find the covariance matrix   C = 1/m XXᵀ  │
└───────────────────────────────────────────┘
             │
             ▼
┌───────────────────────────────────────────┐
│  Find the eigenvalues and corresponding    │
│  eigenvectors of the covariance matrix C   │
└───────────────────────────────────────────┘
             │
             ▼
┌───────────────────────────────────────────┐
│  The eigenvectors are constructed as matrices │
│  according to the size of their eigenvalues, and │
│  the first k rows are taken to form the matrix P │
└───────────────────────────────────────────┘
             │
             ▼
┌───────────────────────────────────────────┐
│  Y=PX which is the data after dimensionality │
│  reduction to K dimensions                 │
└───────────────────────────────────────────┘
             │
             ▼
┌─────────────────────────┐
│           End           │
└─────────────────────────┘
```

Figure 2. Flowchart of PCA dimensionality reduction algorithm.

2.5. Particle Swarm Optimization Algorithm

Weighted data fusion refers to the statistical analysis of multi-sensor data in different times and spaces and then using relevant mathematical methods or practical experience to assign different weights to different sensing data and obtain data fusion values. This includes the weighted average method, Kalman filter, and artificial neural network method. Weighted data fusion aims to obtain a better representation of the features of multi-source data. Weighted data fusion is the fusion of sensor data according to a certain weight, which sets the sensor observation value at $a_i, i = 1, 2, \ldots, n$, where each sensor weighting coefficient is set to $w_{a_i}, i = 1, 2, \ldots, n$ to obtain fused data:

$$\overline{Y} = \sum_{i=1}^{n} w_{a_i} a_i \tag{12}$$

In order to implement the distribution of adaptive weight coefficients among sensor observations, the particle swarm optimization technique and the entropy weight method were introduced in this research. The chaotic mapping algorithm is used to optimize the

particle swarm algorithm because it helps particles jump out of the local optimum and speeds up convergence, which addresses the issue that the particle swarm optimization algorithm is prone to premature convergence to the local optimum and slow convergence at later stages of iteration. This particle's fundamental formula for updating its position and velocity is:

$$X_i = (x_{i1}, x_{i2}, \ldots, x_{iD}), i = 1, 2, \ldots, N \tag{13}$$

$$V_i = (v_{i1}, v_{i2}, \ldots, v_{iD}), i = 1, 2, \ldots, N \tag{14}$$

$$v_{ij}(t+1) = wv_{ij}(t) + c_1 r_1(t)\left[p_{ij}(t) - x_{ij}(t)\right] + c_2 r_2(t)\left[p_{gi}(t) - x_{ij}(t)\right] \tag{15}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{16}$$

where $i = 1, 2, \ldots, N$ denotes the number of particle swarms; j denotes the dimension, P_{ij} denotes the j-th dimension of the individual extreme value of the ith particle; P_{gj} denotes the j-th dimension of the global optimal solution; t denotes the number of iterations of the particle swarms; w is the inertia factor, which is generally taken as the value of 0.5–0.8, and denotes the strength of the algorithm's global optimization seeking ability; and c_1, c_2 is the learning factor, which is generally taken as the value of 0–4.

2.6. Comparison of Fusion Effect

2.6.1. Improved Chaotic Particle Swarm Optimization Algorithm

Mainly in the particle swarm optimization algorithm, chaotic mapping and the entropy weighting method are introduced to achieve the adaptive weighted fusion of vibration signals in the X, Y, and Z directions so that fused signals can characterize more features, the specific process of which is shown in Figure 3.

The specific implementation steps of the improved chaotic particle swarm optimization algorithm are as follows: (where the observation of the sensor is defined as $a_i, i = 1, 2, \ldots, n$. The weighting coefficient of the sensor is also set to $w_{a_i}, i = 1, 2, \ldots, n$.)

(1) Obtaining the original vibration signals: load the original vibration signals in the X, Y, and Z directions to obtain three vectors of length N.

(2) Wavelet packet denoising: 4-layer wavelet packet denoising is performed on the loaded signal to obtain the reconstructed signal in the x, y, and z directions. $a_i, i = 1, 2, \ldots, n$.

(3) Divide the samples: each vector is randomly divided into 200 samples of length 1024 to obtain 600 samples, which are then stored in a 600×1024 matrix.

(4) Entropy weighting method to extract time domain and frequency domain features: for each sample, 14 time domain features and 5 frequency domain features are calculated to obtain a 19-dimensional feature vector. For all 600 samples, a 600×19 feature matrix is formed.

(5) The obtained feature matrix is downscaled using the PCA downscaling algorithm, and the first three principal components are selected according to the contribution rate, constituting a brand new feature matrix of 600×3. This matrix is normalized, and the weight of the feature matrix is calculated using the entropy weight method.

(6) Chaotic particle swarm optimization algorithm: the initial positions and velocities of the particles are optimized using the Logistic chaotic mapping search algorithm, and the fusion weights are iteratively updated using Shannon's direct as the fitness function. According to the optimization results, the optimal fusion weights of the vibration signals in three directions are obtained, $w_{a_i}, i = 1, 2, \ldots, n$. The number of particles is set to 20, the maximum number of iterations for the particle swarm optimization algorithm is set to 50, and the number of iterations of the chaotic mapping is set to 30.

(7) Data fusion: according to the optimal weights, the vibration signals in the three directions are weighted and fused, and the fused data are obtained and saved as a new data set, which is biased and calculated afterward.

$$\overline{Q} = \overline{w}_{a1}\overline{a}_1 + \overline{w}_{a2}\overline{a}_2 + \cdots \overline{w}_{an}\overline{a}_n \tag{17}$$

where Q is the fused vibration signal.
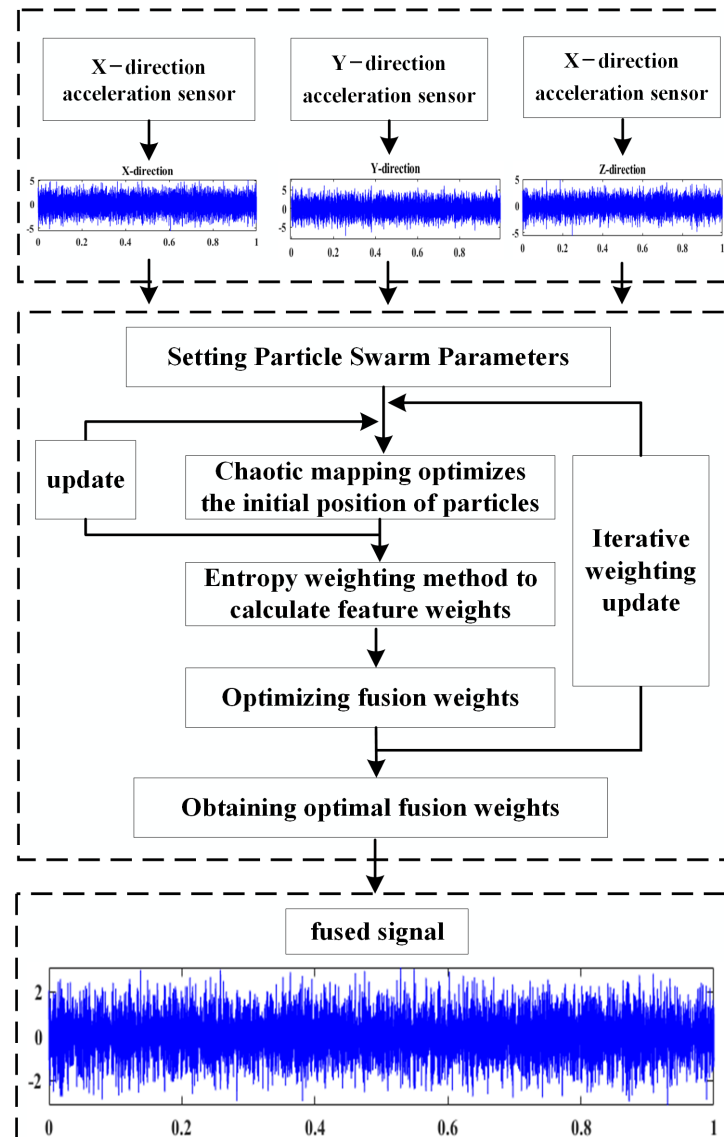
Figure 3. Improved chaotic particle swarm optimization algorithm.

2.6.2. Comparison of Algorithm Fusion Effects

The particle swarm optimization algorithm is primarily used for the adaptive weighted fusion of vibration sensor data in the X, Y, and Z directions to assign the weight coefficients in three directions; therefore, the fused data can more thoroughly and better characterize the effective features of bearings in different states. The comprehensive indexes are primarily established for the analysis of the data-level fusion effects of the following four schemes:

Scheme I: particle swarm optimization (PSO)
Scheme II: particle swarm optimization + entropy weight method (CPSO-EWM)
Scheme III: Chaos mapping + particle swarm optimization (CPSO)

Scheme IV: Chaos mapping + particle swarm optimization + entropy weight method (CPSO-EWM)

The establishment of comprehensive indexes: comprehensive indexes are established based on the characteristics of three indexes: signal-to-noise ratio (SNR), root mean square error (RMSE), and correlation coefficient (Corrcoef).

$$CI = \sum\nolimits_{n=1}^{3} \left(\frac{SNR}{RMSE} + Corrcoef \right), n = [x, y, z] \qquad (18)$$

Table 2 shows that the fused dataset produced by the improved chaotic particle swarm search technique has a superior fusion effect, a greater correlation with the original signal, and a superior fusion of useful aspects for the vibration signals in the *X*, *Y*, and *Z* directions.

Table 2. Comparison of data fusion effects for four schemes.

	PSO	**PSO-EWM**	**CPSO**	**CPSO-EWM**
CI	9.3194	13.9624	11.5777	18.0705

Except for the above comparison on particle swarm algorithms, Table 3 shows that there are some other similar population intelligence optimization algorithms, such as Ant Colony Optimization (ACO), the Artificial Fish Swarm Algorithm (AFSA), and Fish School Search (FSS). The adaptive weighted fusion of isomorphic signals can be achieved by these population intelligence optimization algorithms.

Table 3. Comparison of the effectiveness of common adaptive optimization algorithms.

	PSO	**PSO-EWM**	**CPSO**	**CPSO-EWM**
CI	9.3194	13.9624	11.5777	18.0705

From Table 3, it can be analyzed that the CPSO-EWM algorithm proposed in this paper has some advantages over the other three optimization algorithms. Among them, the fusion effect of ACO and AFSA is close to that of AFSA, and both AFSA and FSS are optimization algorithms based on the behavior of fish populations, though clearly, the fusion effect of AFSA is slightly better.

2.7. Deep Learning Related Modules Introduction

2.7.1. DenseNet Module

DenseNet is a densely connected convolutional neural network whose main purpose is to solve the problems of gradient vanishing and feature repetition during deep network training. The main function of the DenseNet module is to extract effective feature information from input data, and after each convolutional layer, its output is spliced with the output of previous convolutional layers to form a dense connection.

The DenseNet module contains several dense blocks, where each dense block consists of several convolutional layers and pooling layers. In each dense block, all the convolutional layers accept the outputs of all previous convolutional layers and are used as inputs, thus enhancing feature multiplexing and information transfer.

By multiplexing these features, the DenseNet network presents a new structure that not only slows the occurrence of gradient vanishing but also has fewer parameters and is coupled via cross-channel formulas like:

$$x_l = H_l([x_0, x_1, \ldots, x_{l-1}]) \qquad (19)$$

where x_0 and x_l denote inputs to the network and the outputs of layer l, respectively, x_{l-1} is the input to layer $l-1$ of the network, and $H_l(\cdot)$ is the nonlinear transformation operation that acts on layer l.

2.7.2. Transformer Module

The transformer is a neural network model based on the self-attention mechanism for processing sequence data. It mainly consists of two parts: the encoder and the decoder. The Transformer module contains a multi-head self-attention layer, a feed-forward neural network layer, and a residual connection. In signal processing tasks, the Transformer module can be used to extract the important features of elements in the sequence, thus improving the performance of the model. The specific module structure involved is as follows.

(1) Position Encoding: with the introduction of positional encoding, the Transformer model, in order to obtain better parallel computing power, is added to the embedding vector (embedding) of an element as an overall vector by encoding the position of the element in the sequence. Positional coding uses the following functions:

$$PE_{(pos,2i)} = sin\left(pos/10,000^{2i/d\_model}\right) \tag{20}$$

$$PE_{(pos,2i+1)} = cos\left(pos/10,000^{2i/d\_model}\right) \tag{21}$$

where P is the position matrix whose parameters can be updated with the model training process, $P \in R^{n \times d}$.

(2) Multi-head attention: the multi-head attention mechanism used inside the encoder and decoder structures in the Transformer model is obtained by extending the dimensions based on the Scaled Dot-product Attention mechanism.

Scaled Dot-product Attention is a kind of self-attention mechanism, i.e., its own vectors, including Q(query), K(key), and V(value), participate in the computation, and its specific computation is as follows:

$$S_A(Q,K,V) = Softmax(A) \cdot V = Softmax\left(\frac{Q \cdot K^T}{\sqrt{d}}\right) \cdot V \tag{22}$$

where d is the number of dimensions; $S_A(\cdot)$ is the self-attention computation operation; A is the self-attention matrix, $A \in R^{n \times n}$, and n is the sequence length.

The multi-head attention mechanism extends the scaling dot product attention algorithm to multiple dimensions; that is, for the multi-head, after calculating the scaling dot product attention used for multiple information, each result is spliced. The calculation process is as follows:

$$S_{MHA}(\overline{Q}, \overline{K}, \overline{V}) = concat(H_1, H_2, \ldots, H_h) \cdot W \tag{23}$$

$$H_i = S_A(Q_i, K_i, V_i), i = 1, 2, \ldots, h \tag{24}$$

where, Q, K, V is the Q_i, K_i, V_i splicing composition, respectively; W is the linear transformation matrix, $W \in R^{d \times d}$; and $concat(\cdot)$ is the splicing operation.

(3) Residual Connection: the Transformer uses residual connection to enhance the flow of information to improve performance and optimize the training process in combination with the layer normalization operation as follows:

$$R_c = L_N[X + S_{MHA}(x)] \tag{25}$$

where $R_c(\cdot)$ is the residual join operation; $L_N(\cdot)$ is the layer normalization operation; X is the input sequence; and $S_{MHA}(\cdot)$ is the use of multiple attention mechanisms.

(4) Data enter a fully connected network made up of two linear transformation layers and one nonlinear activation layer after being output from the multi-attention layer. The activation function in this network uses a linear rectification function.

$$Net_{FFN} = W_2 \cdot f(W_1 \cdot X) \tag{26}$$

where $Net_{FFN}(\cdot)$ is the feedforward network; W_1, W_2 is the parameter of each of the 2 linear layers; and $f(\cdot)$ is the nonlinear activation function.

(5) Max-pooling: the Transformer module's encoder ends with the introduction of the pooling layer downsampling function. The pooling procedure, in which the pooling layer adopts the maximum pooling can lower the size of the feature vectors and the danger of overfitting.

$$y_a = max\left(r_a^{n \times n} u(n, n)\right) \tag{27}$$

where y_a is the output feature of region a; $r_a^{n \times n}$ represents the $\alpha - th$ region of size $n \times n$; and $u(n, n)$ represents the window function of size $n \times n$.

2.7.3. Introduction to the DAT feature extraction model

The DAT deep learning model (DenseNet and Transformer, DAT) is a tandem combination of DenseNet and Transformer modules, which can realize more complex feature extraction and transformation; this structure is shown in Figure 4. The deep feature extraction model uses a tandem combination of DenseNet and Transformer for one-dimensional signal feature extraction and has the following roles and advantages:

(1) Increase feature extraction's effectiveness: the Transformer, on the one hand, uses the self-attention mechanism, which is able to better capture key information in the sequence and improve the accuracy and efficiency of feature extraction. DenseNet, on the other hand, has the characteristic of dense connection, which can more fully utilize low-level features for classification and improve the efficiency of feature extraction.

(2) Increase the model's capacity for generalization: Transformer and DenseNet both possess excellent feature extraction and generalization capabilities, enabling them to deal with complicated changes and noise interference in the bearing signal and increase the model's capacity for generalization.

(3) Make the most of the bearing vibration signal's time series properties. The bearing signal is a type of time series signal and contains a few time series features. Both DenseNet and Transformer can fully utilize time series features to extract more thorough and precise feature representations, resulting in better classification of the signal.
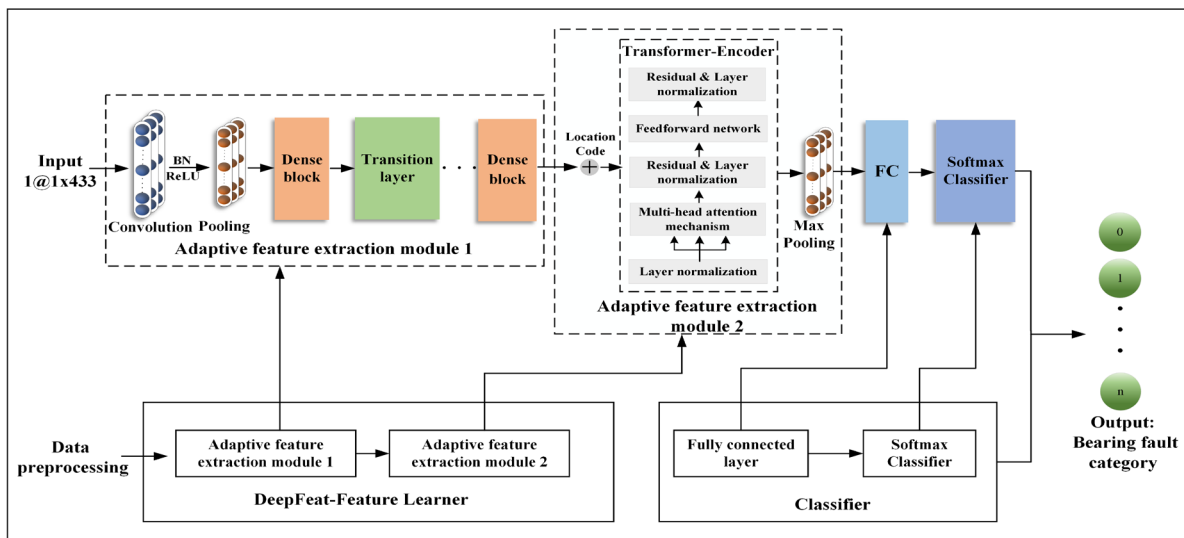

Figure 4. Overall structure of the DAT-based diagnostic model.

In conclusion, the feature extraction of bearing one-dimensional signals using DenseNet and Transformer in tandem can significantly increase the model's classification accuracy, generalizability, and interpretability, making it ideal for challenging signal classification applications.

Figure 4 depicts the overall structure of the DAT-based fault detection model, which is composed of a feature learner and a deep learning classifier and is implemented as follows: (1) preprocessing and the sample expansion of sensor measurement data is performed (see Section 2.3); (2) Fourier variation is used to transform the preprocessed one-dimensional vectors into a spectrum of 1433 before being input into the DAT model; (3) in the learning feature, two deep feature extraction processes are used. the convolutional layer, pooling layer, dense block (DenseBlock), transition layer, etc., make up the main components of the DenseNet dense connected network (deep feature extraction module 1), and the multi-head self-attention layer, feed-forward neural network layer, residual connection, and other model structures make up the main components of the Transformer module (deep feature extraction module 2). (4) The extracted features are input into the feature classifier (the fully connected network layer and residual connection) and other model structures; Table 4 displays the unique DAT diagnostic model parameters.

Table 4. DAT network parameters.

Model Name	1D-DAT	
	Structure Type	**Convolution Kernel**
Input layer	1D FFT spectrum	—
Convolution layer	Conv	1×7
Pooling layer	Max-pooling	1×3
Adaptive Features Extraction Module 1	Dense block $-1: \begin{Bmatrix} BN - Relu - Conv \\ BN - Relu - Conv \end{Bmatrix} \times 1$	$\begin{Bmatrix} 1 \times 1 \\ 1 \times 3 \end{Bmatrix} \times 1$
	Transition layer-1:BN-Relu-Conv-Pooling	$\begin{Bmatrix} 1 \times 1 \\ 1 \times 2 \end{Bmatrix} \times 1$
	$\cdots\cdots$	$\cdots\cdots$
	Transition layer-2:BN-Relu-Conv-Pooling	$\begin{Bmatrix} 1 \times 1 \\ 1 \times 2 \end{Bmatrix} \times 1$
	Dense block $-3: \begin{Bmatrix} BN - Relu - Conv \\ BN - Relu - Conv \end{Bmatrix} \times 1$	$\begin{Bmatrix} 1 \times 1 \\ 1 \times 3 \end{Bmatrix} \times 1$
Adaptive Features Extraction Module 2	Position code$\times 1$	—
	Encoder$\times 1$	—
	Encoder$\times 1$	—
	Max-pooling	$\times 1$
Fully connected layer	FC	—
Output layer	SoftMax	—

3. Model Setup and Training

3.1. Condition Monitoring Model Introduction

A data-level fusion method is proposed in this paper based on multi-source isomorphic sensors to monitor the operational status of rolling bearings, as shown in Figure 5. This procedure includes the decomposition of wavelet packets, denoising, the reconstruction of the vibration signal, feature extraction in the time and frequency domains, and dimension reduction using the PCA algorithm. The chaotic particle swarm algorithm is used with the entropy weighting method to produce the initial weights and global optimal weights. Last but not least, the Transformer module is shown in order to build the DAT feature extraction model and enhance the precision and effectiveness of feature extraction. This technique can successfully handle the requirement for tracking the operational status of machine tool spindle bearings under challenging operating conditions.
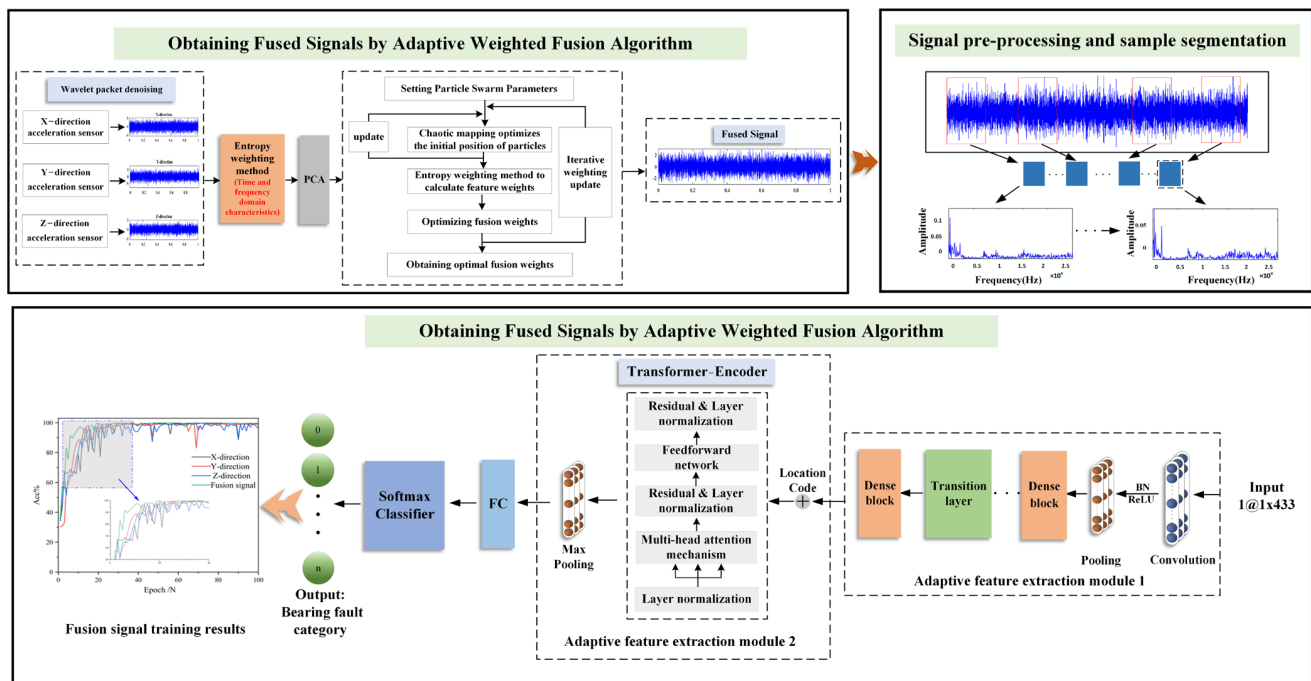
Figure 5. Overall structure of the condition monitoring model.

3.2. Data Pre-Processing

Experimental dataset S1: Open-source data on rolling bearing faults were published by Jiangnan University's experimental platform to evaluate and study the model for identifying rolling bearing faults. The experimental rolling bearing fault diagnosis system for wind turbines at Jiangnan University is depicted in Figure 6. Rolling bearing vibration signals were collected at speeds of 600, 800, and 1000 rpm with a constant rotational speed of 1 krpm, a sampling frequency of 50 kHz, and a sampling time of 10s [37]. Bearing failure was man-made through the wire cutting technology, respectively, in the bearing inner ring, outer ring, rolling body, and the processing of 0.3 * 0.05 mm (width * depth) tiny wounds, as can be seen from the waveform diagram. The data of various states are difficult to distinguish directly from this waveform diagram.

Figure 6. Rolling bearing failure test platform of Jiangnan University.

Each set of experiments involved three different fault conditions (rolling body damage, inner ring damage, and outer ring damage) as well as one type of normal condition. Table 5 for the speed of 600 r/min in the experimental dataset describes the type of fault conditions, as shown in Figure 7, for the experiments according to fan speed.

Table 5. Experiment 1 (600 r/min) data set.

Experimental Conditions	Training Set: Validation Set: Test Set	Labels
Inner ring failure	280:80:40	IF
Outer ring failure	280:80:40	OF
Ball Failure	280:80:40	BF
Normal	280:80:40	Normal
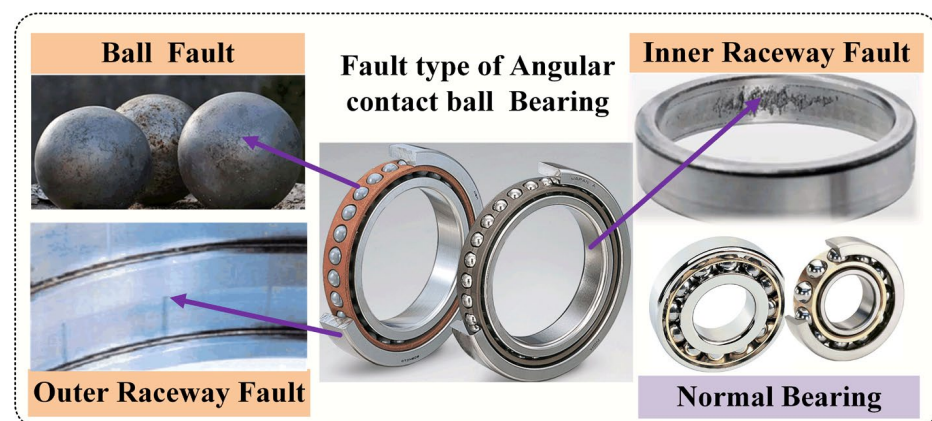
Figure 7. Distribution of bearing condition types.

Numerous high-frequency and low-frequency components with varying sensitivity levels can be found in the time domain of the vibration signal, which is used to diagnose bearing defects. In order to analyze the time-domain signals more effectively, it is necessary to convert them into frequency-domain signals. As depicted in Figure 8, four bearing signals—BF, IF, OF, and normal—under a rotation speed of 600 r/min were taken for spectrum analysis, and 1024 data points were taken as a sample for FFT transformation. It can be found that the normal state of the bearing, the fault of the rolling element, the fault of the inner ring, and the fault of the outer ring have obvious differences in the amplitude of the whole stage of the spectrum, which can effectively identify the frequency component of the signal and provide useful information for the application of signal feature extraction, classification, and diagnosis.

Experimental dataset S2: To further investigate the monitoring function of this method during the double-bearing operation, an unbalanced bearing load test rig was developed, as shown in Figure 9. A non-balanced bearing load test platform was designed and manufactured, and the monitoring function of this technology was investigated during the bearing operation. The test platform, as illustrated in Figure 9, consists of a motor, precision spindle, roll bearing, and acceleration sensor with a maximum speed of 10,000 r/min. A flexible coupling connects the mechanical spindle to the electric spindle, and the motor action is regulated by a servo control system. The hardware consists of a motorized spindle, a rotational accuracy test device, a data collector, a computer, and other components.

The platform employed four NSK 7014C angular contact ball bearings, with the positions of F1, F2, and F3 evenly spaced at 120°. Preloads of different sizes were set to determine the operating conditions of bearings, including light (C2), medium (C4), and heavy (C6) loads. The bearings were mounted back-to-back, and the fixed speed of the test platform was set at 4000 r/min with a sampling frequency of 8192 Hz. The parameters of the bearings are shown in Table 6.
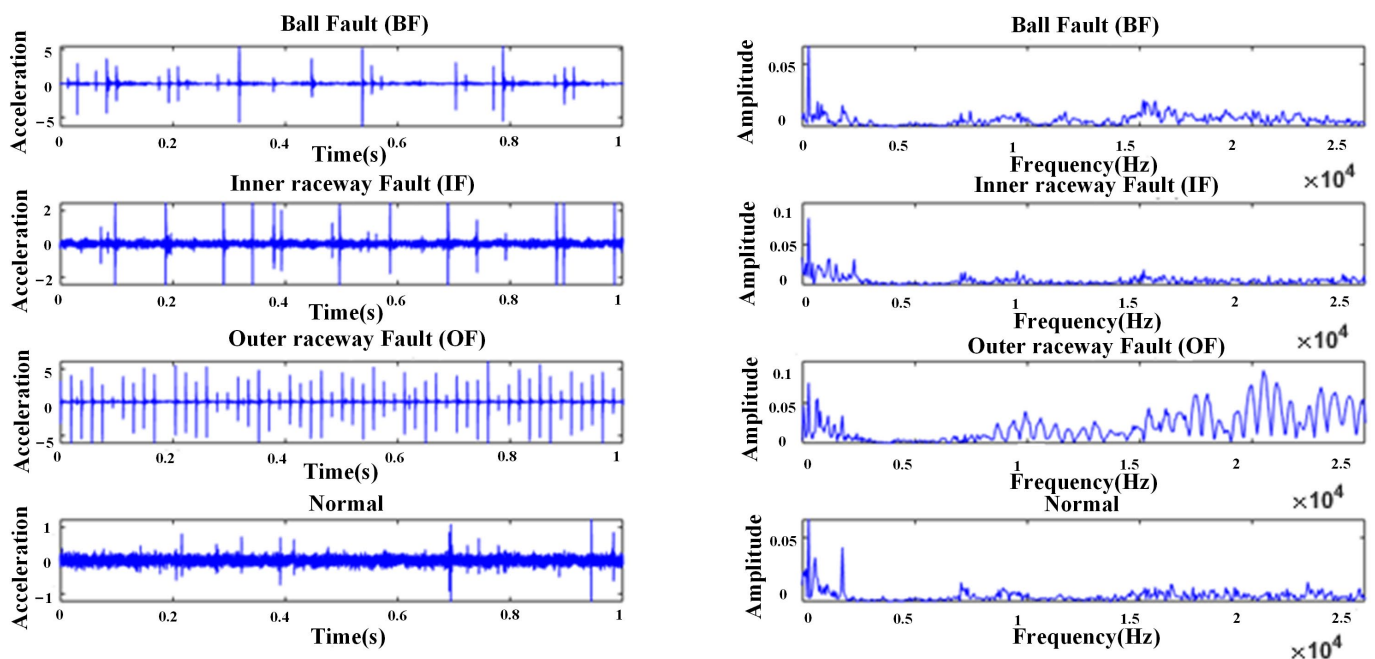
Figure 8. Time-domain waveforms and spectrograms of bearing data at Jiangnan University.

The purpose of building the test platform was to distinguish the working condition of bearings under an unbalanced operation so as to detect the bearing failures caused by wrong assembly or processing in real-time. Due to the limited conditions of the laboratory, the current test platform can only be used to verify the effectiveness and accuracy of the condition monitoring method and cannot simulate the corresponding bearing fault state under different loads.

3.2.1. Data Normalization

Normalized preprocessing is a commonly used data processing method. These data are mapped to a specific interval by applying a linear transformation to the data and commonly mapping the data to the interval [0, 1]. If sample data are supposed to be $X = \{x_1, x_2, ..., x_n\}$, the normalized transformation formula is as follows:

$$y_i = \frac{x_i - min\{x_j\}}{max\{x_j\} - min\{x_j\}} \tag{28}$$

where y_i is the result of normalization, x_i is the $i-th$ sample data, $max\{x_j\}$ is the maximum value of the sample data, and $min\{x_j\}$ is the minimum value of the sample data.

3.2.2. Overlapping Sampling

In the realm of data-driven deep learning, having enough big training samples is essential to increase model accuracy and significantly lower overfitting. As illustrated in Figure 10, we used overlapping sampling with a moving sliding window to increase the number of training samples, which can better capture changes and patterns in time series data. This method can avoid the problem of signal loss caused by equidistant sampling and sampling, thus improving the training effect and generalization ability of the model.
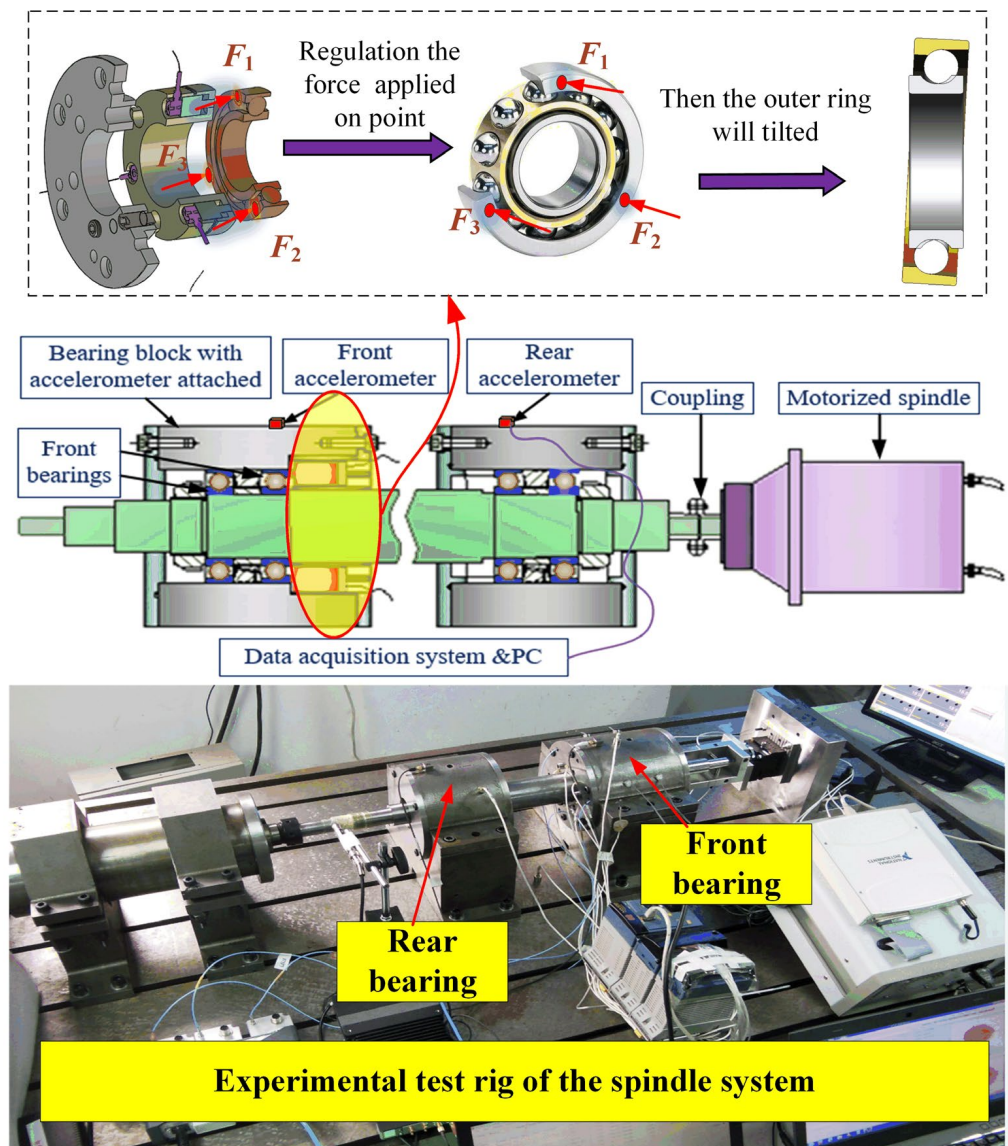
Figure 9. Structure of non-uniform preloading test stand.

Table 6. Parameters of NSK 7014C angular contact ball bearings.

Inner Ring Diameter/mm	Outer Ring Diameter/mm	Thickness/mm	Dynamic Load/KN	Static Load/mm
70	100	20	47	43

In order to avoid the loss of detailed features, overlap sampling is used to extend the original data samples. By adjusting the parameters such as offset, data length, expansion multiplier, and the number of samples, the detection of the model performance under different numbers of samples can be achieved. This can be achieved by flexibly adjusting the sampling parameters so as to optimize the training and prediction ability of the model.
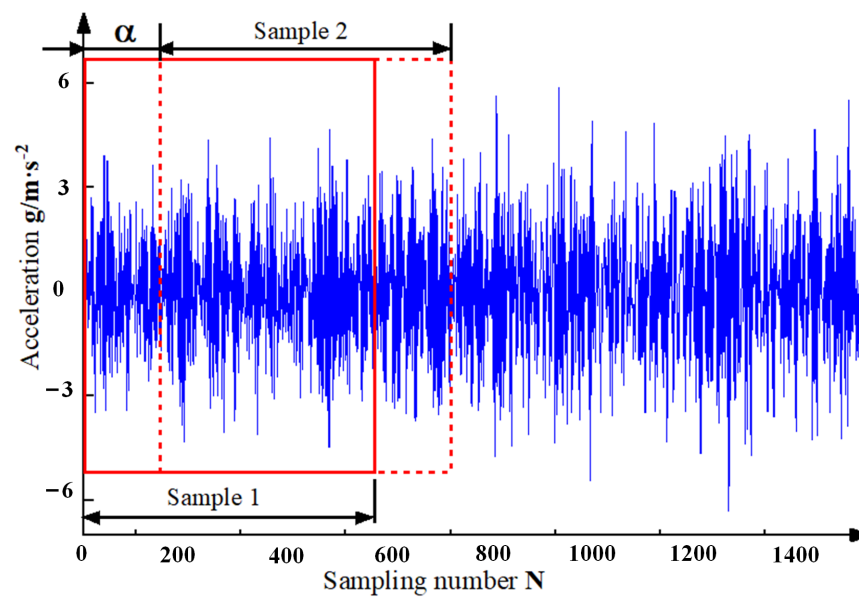
Figure 10. Schematic diagram of data enhancement.

3.3. DAT Model Hyperparameter Settings

The AdamP backpropagation algorithm proposed by ByeonghoHeo et al. [38] was selected as the optimization method, which improved the performance of small-batch training, reduced the risk of overfitting, and delayed the attenuation of effective step size, thus training the model at a barrier-free speed, retaining many advantages of the Adam algorithm, such as adaptive learning rate adjustment and momentum term.

The rolling bearing service condition monitoring model based on DAT is based on features that diagnose working conditions, with data under different working conditions mainly classified and recognized. The Cross-entropy loss function was chosen as the base function, and some improvements were made.

The Cross-entropy loss function is formulated as follows:

$$loss = -\sum\nolimits_{\theta} p(\theta)\log q(\theta) \tag{29}$$

where θ denotes the learning parameter; $p(\theta)$ and $q(\theta)$ are the correct probability and prediction probability of the label.

The K-L scatter of $p(\theta)$ and $q(\theta)$ is as follows:

$$KLD = \sum\nolimits_{\theta} p(\theta)\log \frac{p(\theta)}{q(\theta)} \tag{30}$$

When calculating Cross-entropy loss using KL scatter, it is often necessary to transform the true labels into probability distributions. The traditional approach is to use one-hot coding, where only one element is 1, and the rest are 0, indicating the category to which the true label belongs. However, this approach may lead to the overconfidence or oversensitivity of the model in the presence of noise and uncertainty.

In order to reduce the impact of label noise and uncertainty in the model, this paper uses smoothed target labeling. By smoothing the target labels, we can better adapt to complex data distributions and noise situations and, thus, obtain a more accurate loss function of Ce.

Figure 11 shows that the iteration effect of the experiment is significantly improved when using Ce_loss compared to the iteration curve of the model under Cross-entropy loss. The accuracy of the training set of this paper's method reached 99% after 10 iterations, while the accuracy of Cross-entropy loss reached 99% after 85 iterations. The experiment at 600 r/min under the S1 dataset was chosen for analysis. The experimental results demon-

strate that the strategy suggested in this research can significantly enhance the model's training performance while shortening the training period. Through the experimental analysis, the parameters of the model were set, as shown in Table 7.
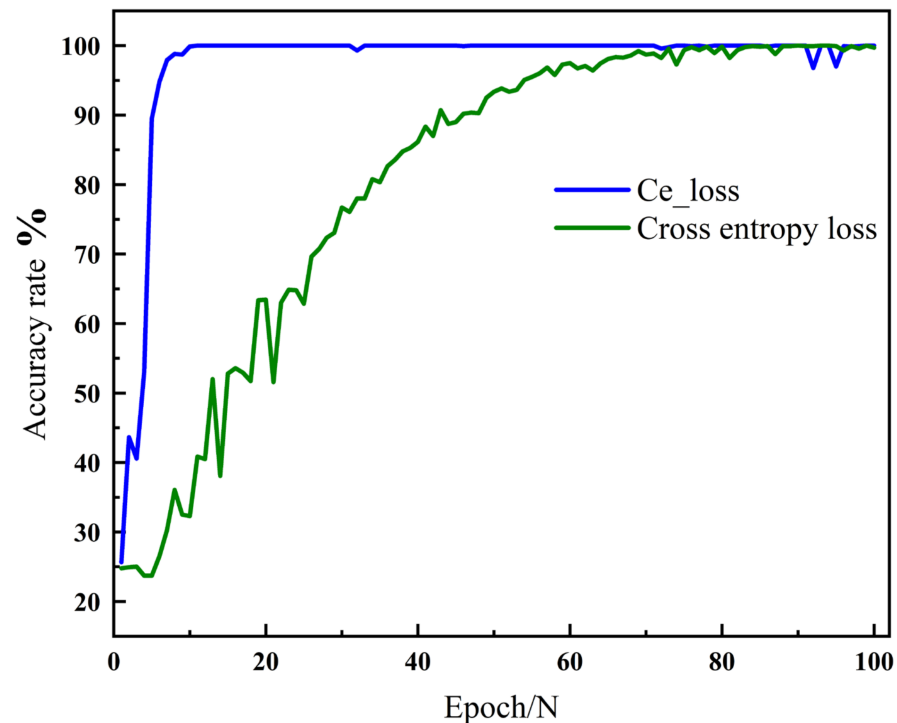

Figure 11. Model iteration curves under 600 r/min experiment with different loss functions.

Table 7. Model parameters (experimental data set S1).

Parameter Category	Parameter Setting
Optimizer	AdamP
Loss function	Ce_loss
Number of iterations	100
Initial learning rate	0.001
Smoothing	0.1
Batch Size	64

3.4. Model Training

Each group of tests was repeated five times to assess the accuracy and stability of the proposed model for rolling bearing failure diagnosis with the average accuracy and greatest accuracy of the experiments provided in Table 8. On the rolling bearing fault data collected by Jiangnan University's rolling bearing fault diagnostic platform, fault diagnosis was performed using the deep learning framework PyTorch (see Table 5 for details).

Table 8. Fault diagnosis results of rolling bearing based on DAT model.

Data Sets	Experimental Setup	Number of Iterations	Training Time/s	Average Acc/%	Maximum Acc/%
Jiangnan University	600 r/min	100	57	99.375%	99.756%
	800 r/min	100	62	99.583%	99.625%
	1000 r/min	100	58	99.285%	99.423%

As shown in Figure 12, the DAT model's greatest accuracy on the validation set at 600 r/min was 99.8%, and after five iterations, training accuracy was 90%. The model was

more stable throughout the training process, and there was no abrupt change in accuracy, which suggests that this model has a strong ability to generalize, is highly robust, is able to capture data in real patterns, and is not easily disturbed by noise and outliers. This can be seen by analyzing the iteration curves in the Figure below.
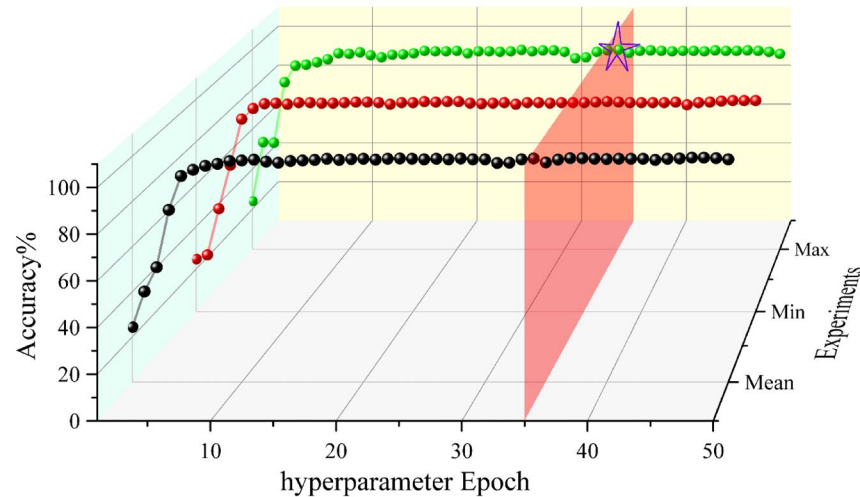
Figure 12. Model training at 600 r/min rotational speed.

Figure 13 shows the confusion matrix of the model under the rolling bearing fault diagnosis platform of Jiangnan University, which was used to evaluate the performance of the classification model and helped us gain a more comprehensive understanding of the classification effect of the model in different categories.

Figure 13 shows that the DAT model achieves the highest accuracy of 99.5% in five experiments on the experimental dataset of 600 r/min at Jiangnan University. By analyzing the confusion matrix, it was found that 2% of the IF600 (Inner ring failure) were incorrectly predicted as BF600 (Ball failure), 2% of the Normal600 (Normal) were incorrectly predicted as IF600 (Inner ring failure), and the rest were correctly classified.

Under the same experimental conditions, the proposed DAT fault diagnosis model was compared with Transformer, DenseNet-LSTM, CNN-LSTM, DenseNet, and other models in the comparison experiments. For the experimental data of rolling bearing faults in Jiangnan University, the average accuracy comparison results of the above different models are shown in Figure 14.

On the basis of DenseNet, the DAT model introduces the Transformer module, in which the Transformer employs the self-attention mechanism, which can better capture key information in the sequence, improve the accuracy and efficiency of feature extraction, and make the DAT model have a better fault diagnosis effect. Table 9 shows that the average accuracy of the DAT model on the experimental data of Jiangnan University's rolling bearing defects is higher than that of the other four models. DenseNet-LSTM adds the LATM layer to the DenseNet network, whereas the DAT model adds the Transformer module to the DenseNet network where both models have higher diagnostic accuracy, but the DAT model has a slightly higher diagnostic accuracy. In terms of learning features, the DAT model is slightly inferior to the DenseNet-LSTM model, implying that the self-attention mechanism is superior to LSTM, which is suited for temporal signal prediction.

In summary, it can be seen that the DAT fault diagnosis model has higher fault diagnosis accuracy and better stability than Transformer, DenseNet-LSTM, CNN-LSTM, DenseNet and other models.
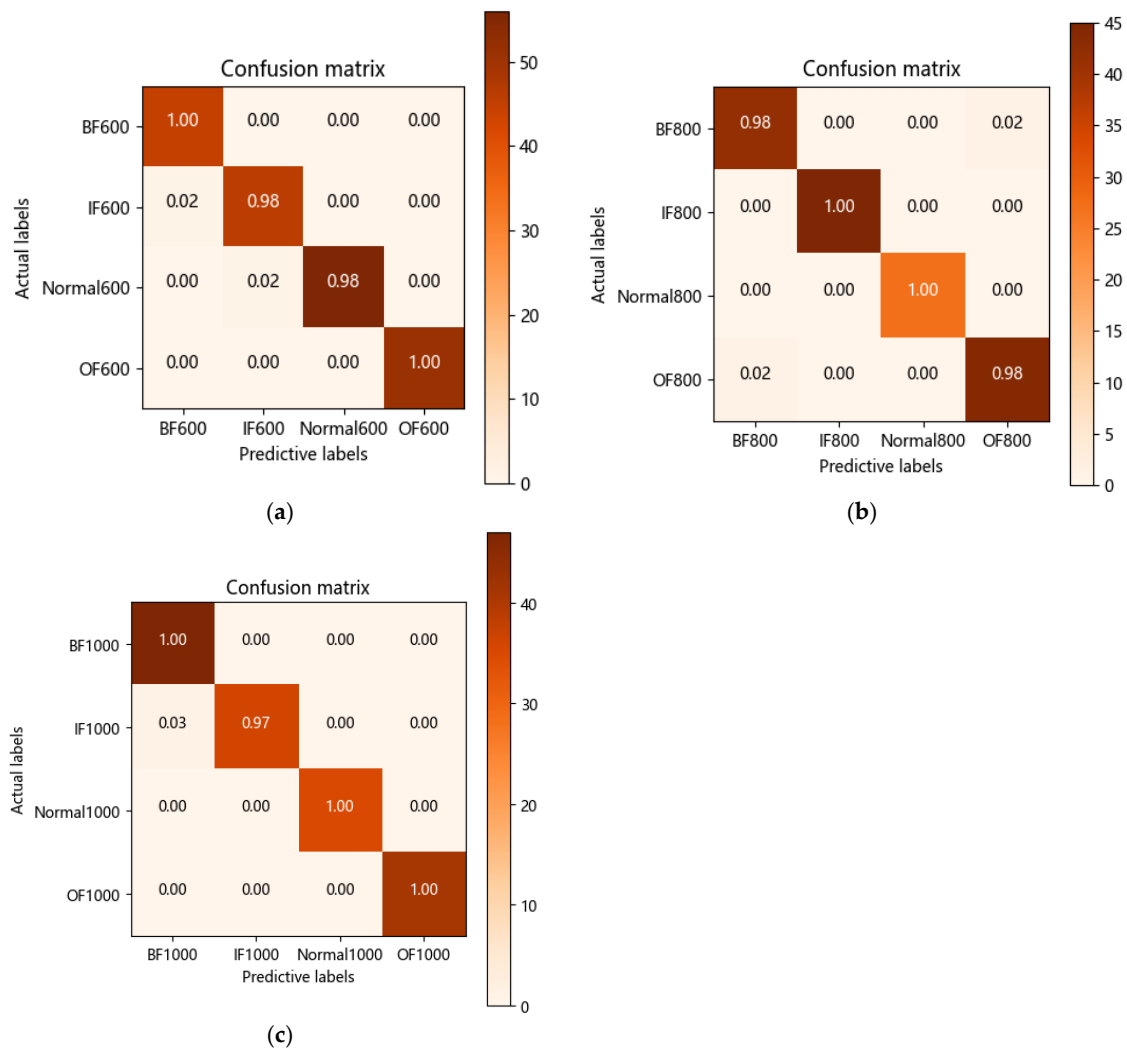
Figure 13. Confusion matrix based on DAT modeling. (**a**) 600 r/min (**b**) 800 r/min (**c**) 1000 r/min.
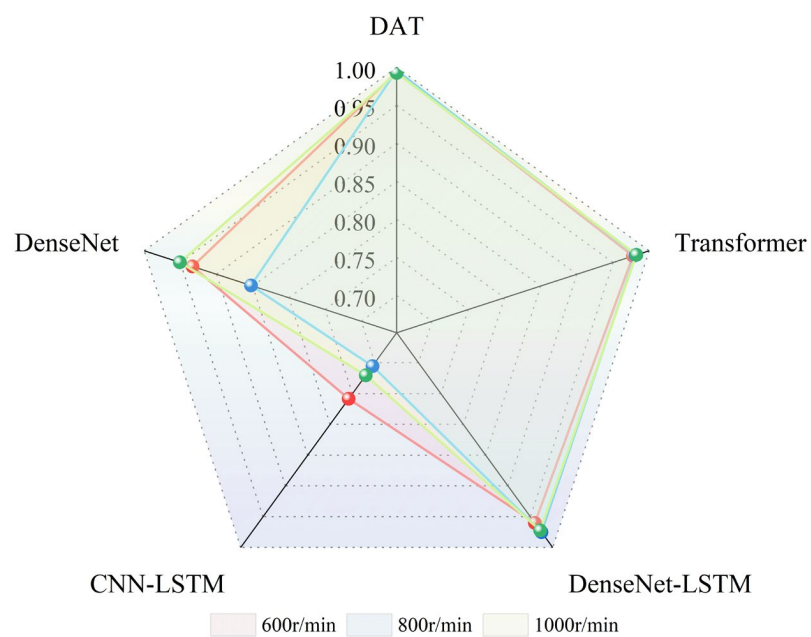

Figure 14. DAT model and its comparison model with average accuracy at different rotational speeds.

Table 9. Average accuracy of DAT model and its comparison model under 5 experiments.

Experiments	DAT	Transformer	DenseNet-LSTM	CNN-LSTM	DenseNet
600 r/min	99.275%	97.725%	96.768%	75.833%	93.333%
800 r/min	99.583%	98.583%	97.525%	70.512%	85.233%
1000 r/min	99.158%	98.375%	97.245%	72.012%	95.076%

4. Fusion Data Testing

The experimental data set S2 was measured using the non-uniform load operation fault simulation test platform. The data of the light load (C2), medium load (C4), and heavy load (C6) under the positions of F1, F2, and F3 were collected, and the experimental data sets were divided as shown in Table 10. Bearing vibration data can be classified into nine types. The experiment was divided into three groups, each of which included 840 samples in the training set, 240 samples in the test set, and 120 samples in the verification set.

Table 10. Experimental data set (experimental data set S2).

Experimental Setup	Signal Type	Training Set	Validation Set	Test Set
Experiment 1 (F1 position)	F1(C2) = 400 N	840	240	120
	F1(C4) = 800 N	840	240	120
	F1(C6) = 1200 N	840	240	120
Experiment 2 (F2 position)	F2(C2) = 400 N	840	240	120
	F2(C4) = 800 N	840	240	120
	F2(C6) = 1200 N	840	240	120
Experiment 3 (F3 position)	F3(C2) = 400 N	840	240	120
	F3(C4) = 800 N	840	240	120
	F3(C6) = 1200 N	840	240	120

In the time domain, by converting the signal to the frequency domain, we could decompose this signal into components of different frequencies and analyze the amplitude of each frequency component. In Figure 15, we selected the bearing signal under the C2 operating condition at the F2 position for spectral analysis. The 1024 data points were taken as a sample, and FFT (Fast Fourier Transform) was transformed to obtain the corresponding spectrogram. From the spectrograms, it can be observed that the spectrograms of all four sets of vibration data had the maximum amplitude variation at 3044 Hz. This reflects the major frequency components of the signal in the frequency domain. In isomorphic data fusion, the low-frequency component often represents low-frequency noise or a slow vibration, which is usually as small as possible. By analyzing the spectrogram, it can be seen that the fused signal had a smaller low-frequency component amplitude at 3044 Hz, indicating that the fusion effect of the isomorphic data fusion method proposed in this paper is clearer.

The experiment's input sample length was set to 1×1024, its Fourier transform was run to choose a 1×433 spectrum as the model's input, the batch size was 64, the number of training iterations was 100, the learning rate was 0.05, and Adam was chosen as the optimizer.

As can be seen from Figure 16, the accuracy curve of the fused signal is relatively stable, reaching 94% after 10 iterations, and the accuracy of the model is stable at 99.34% after 50 iterations. The local zoom-in graph shows that the iterative process of the fused signal is more stable, indicating that the fused signal has more effective features.

According to Table 11, it can be seen that the accuracy of the fused data for the fault identification on the DAT model is slightly higher than that of using a single-direction vibration signal, and the accuracy of the fused data is more stable compared to the unisex signal. In order to judge the classification performance of the training results intuitively, a confusion matrix was used to present them visually, as shown in Figure 17, where the

horizontal and vertical axes labels represent the three working conditions of light load (C2), medium load (C4) and heavy load (C6).
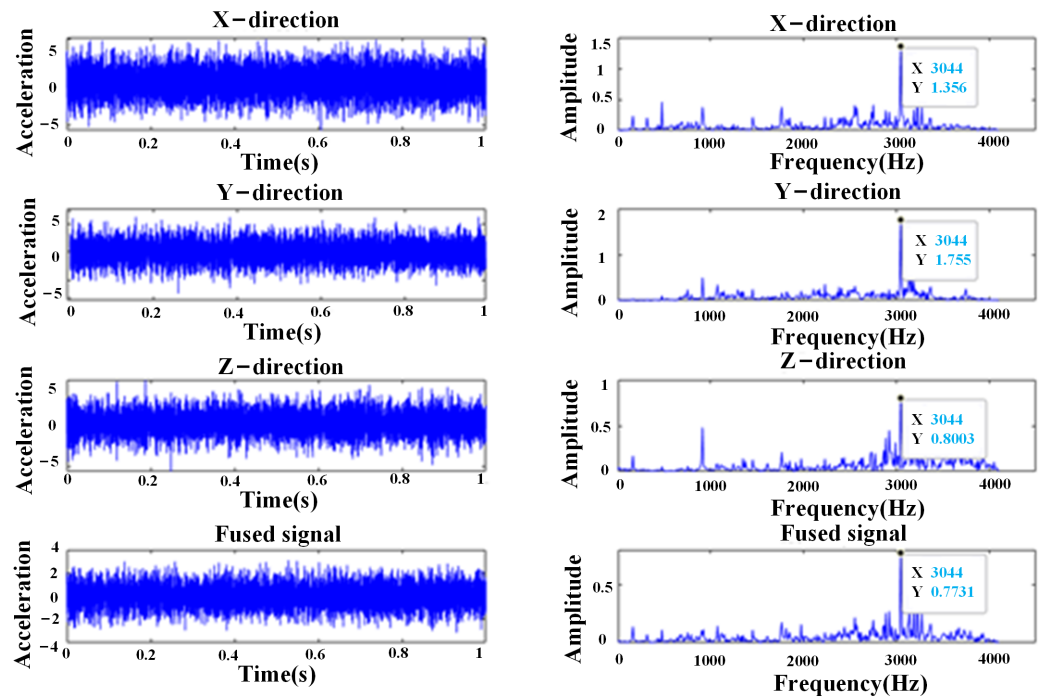

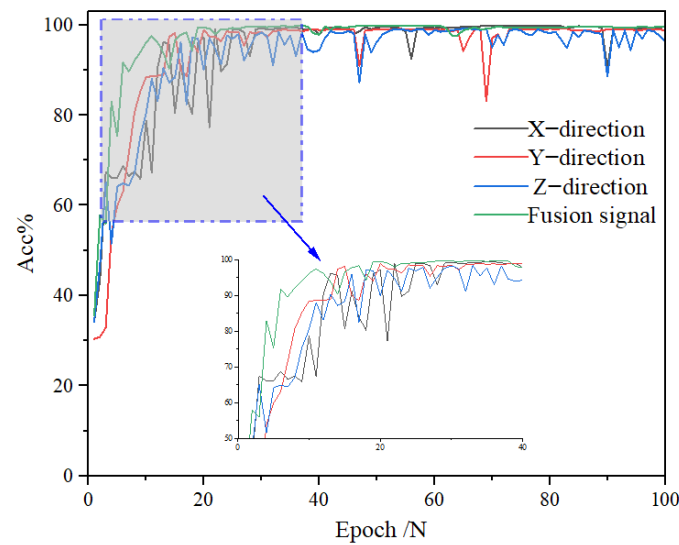**Figure 15.** Signal analysis of X, Y, Z and fused signal at F1 position.

Figure 16. Accuracy curve of fused signal with X, Y and Z signals.

Table 11. Results of control experiments (50 iterations and average accuracy over 5 experiments).

Experimental Setup	Fusion Data		X-Direction Average/Acc%	Y-Direction Average/Acc%	Z-Direction Average/Acc%
	Avg/Acc%	Max/Acc%			
F1 position	99.76%	100%	99.15%	98.56%	97.72%
F2 position	99.82%	100%	98.66%	97.66%	99.17%
F3 position	99.92%	100%	98.52%	99.23%	99.40%

For visualization, the high-dimensional features collected from the DAT model's input layer and final hidden layer were mapped into three-dimensional feature vectors. Figure 16

depicts the visualization results of the features with the best accuracy in five experiments, where the numerical point reflects the fault diagnosis efficiency of the algorithm utilized in this study. Figure 18 shows that among X, Y, Z, and the fusion signal, the fusion signal had the best fault classification impact, showing that the DAT model's classification effect was superior.

(a)

(b)

(c)

(d)
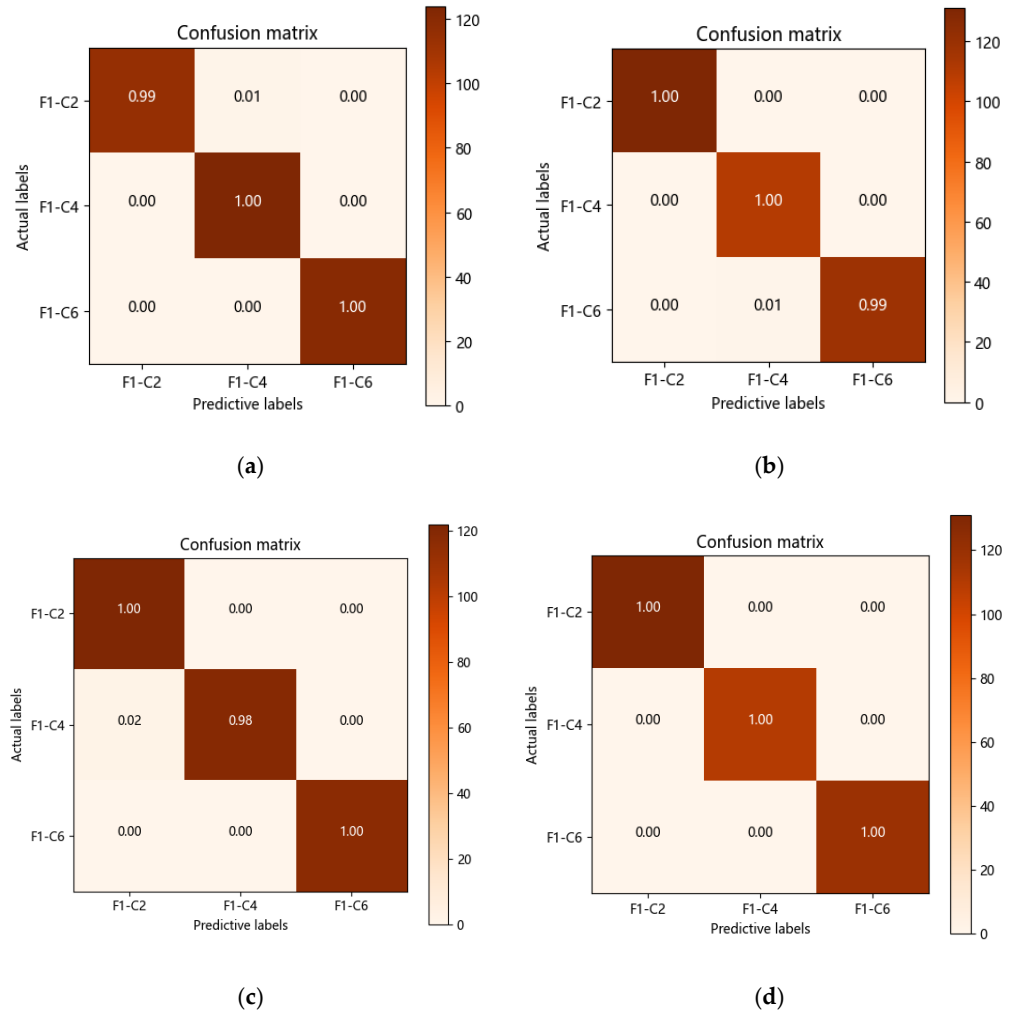
Figure 17. Confusion matrix for fault classification of the first set of experiments. (**a**) X direction confusion matrix (**b**) Y direction confusion matrix. (**c**) Z direction confusion matrix (**d**) Fusion signal confusion matrix.

(a)

Figure 18. *Cont.*

(b)

(c)

(d)

Sample Categories: ● F1-C2 ● F1-C4 ● F1-C6
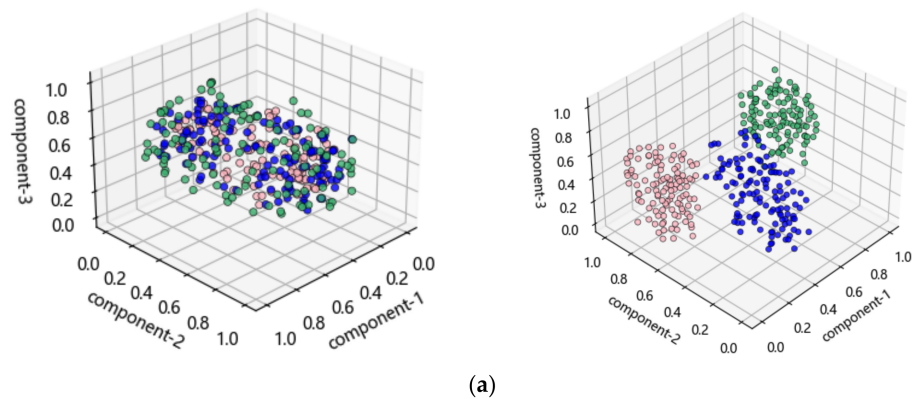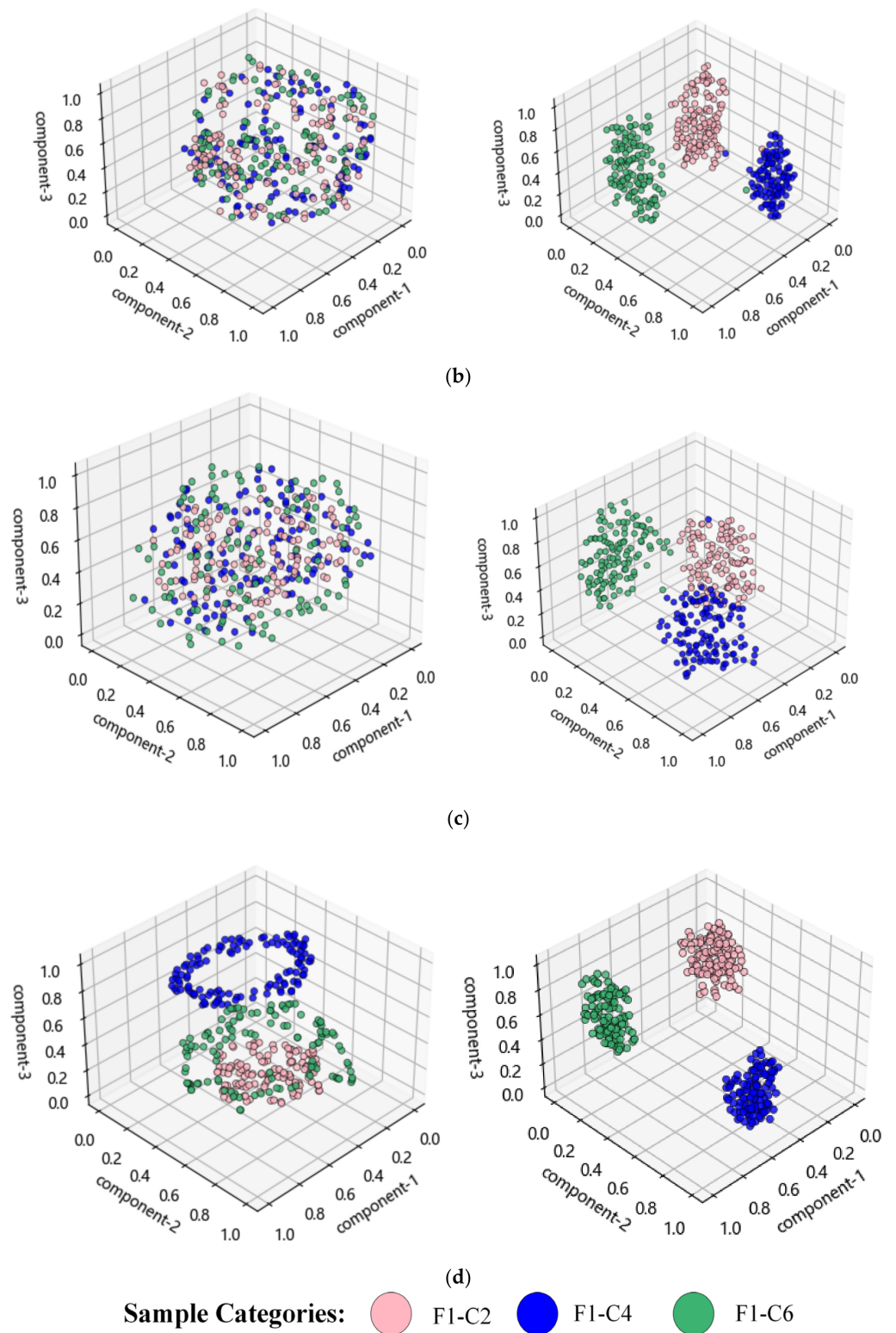
Figure 18. Visualization of input and output layer features for the first set of experiments. (**a**) X direction (Max-99.72%). (**b**) Y direction (Max-99.72%). (**c**) Z direction (Max-99.58%). (**d**) Fusion signal (Max-100%).

Figures 17 and 18 show that in five tests from the first set of experimental datasets of S2 obtained from the bearing non-uniform load test platform, the DAT model similarly attained a minimum accuracy of more than 99%. In total, 1% of the samples from F1-C2 (low load) in X-direction tests were wrongly projected as F1-C4 (medium load), according

to the confusion matrix and the output feature downscaling visualization plot, while the remaining samples were correctly classified. Combining information from various isomorphic acceleration sensors can boost redundancy and boost the accuracy of defect finding. Data from other sensors are still available in the event of a sensor failure or abnormality, maintaining the stability of the system and the accuracy of fault identification. By examining the fused signals' X, Y, and Z directions as well as the classification outcomes, it was discovered that the fused signals' highest degree of classification accuracy is 100%, their original features are more distinct from one another, and there was no classification abnormality when compared to the other three signal groups.

5. Conclusions

A data-level fusion method based on multi-source isomorphic sensors is proposed in this paper to monitor the working condition of rolling bearings. The vibration data in the X, Y, and Z directions of raw data were fused firstly using a chaotic particle swarm optimization algorithm. Then, a DAT feature extraction model was built to extract the deep features of the fused signals. Finally, the overall iterative performance of the model was improved using the AdamP optimization algorithm and the improved Ce_loss loss function, reaching the following conclusion.

- The data-level fusion method of multi-source homogeneous sensors is proposed by fusing data from different sensors. Information of multiple dimensions can be obtained, which makes the perception of the target object or the environment more comprehensive and accurate and enhances time-domain continuity alongside the consistency of data, which can be enhanced as well as the fault tolerance and robustness of the system.
- A DAT deep feature extraction model can be constructed to monitor the working condition of spindle bearings, which can recognize the bearing faults and unbalanced loads.
- Through the AdamP optimization algorithm and the improved Ce_loss loss function, the iterative performance of the proposed model can be drastically improved, and the steady state can be reached faster.
- This study validates the fusion performance of isomorphic signals and the diagnostic performance of the model. In the future, we plan to apply the DAT model to other components of the spindle system and migrate it to other fields for validation. This could expand the applicability of the model and increase its value in practical engineering applications.

Author Contributions: Conceptualization, Y.Z.; methodology, Y.Z., Y.L.; software, Y.L.; validation, Y.L., M.Y. and X.F.; formal analysis, Q.Z.; investigation, Y.L.; resources, L.K.; data curation, Y.L.; writing—original draft preparation, Y.L.; writing—review and editing, Y.Z.; visualization, Q.Z.; supervision, Y.Z.; project administration, L.K.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: Not applicable.

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

References

1. Jing, L.; Zhao, M.; Li, P.; Xu, X. A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox. *Measurement* **2017**, *111*, 2–4. [CrossRef]
2. Jeschke, S.; Brecher, C.; Meisen, T.; Özdemir, D.; Eschert, T. *Industrial Internet of Things and Cyber Manufacturing System*; Springer: Cham, Switzerland, 2017; pp. 3–19.
3. Yin, S.; Li, X.; Gao, H.; Kaynak, O. Data-based techniques focused on modern industry: An overview. *IEEE Trans. Ind. Electron.* **2014**, *62*, 657–667. [CrossRef]

4. Ma, S.; Yin, Y.; Chao, B.; Yan, K.; Fang, B.; Hong, J. A Real-time Coupling Model of Bearing-Rotor System Based on Semi-flexible Body Element. *Int. J. Mech. Sci.* **2023**, *245*, 108098. [CrossRef]

5. Fang, B.; Zhang, J.; Hong, J.; Yan, K. Research on the Nonlinear Stiffness Characteristics of Double-Row Angular Contact Ball Bearings under Different Working Conditions. *Lubricants* **2023**, *11*, 44. [CrossRef]

6. Zhang, Y.F.; Li, Y.H.; Kong, L.F.; Li, W.C.; Yi, Y.J. Rolling bearing condition monitoring method based on multi-feature information fusion. *J. Adv. Manuf. Sci. Technol.* **2022**, *3*, 2022020. [CrossRef]

7. Zhou, J.; Zhu, J.W. Research on machine learning classification problems and algorithms. *Software* **2019**, *40*, 205–208.

8. Zhang, R.; Wang, Y.B. Research on machine learning and its algorithms and development. *J. Commun. Univ. China Nat. Sci. Ed.* **2016**, *23*, 10–18.

9. Wu, X.M.; Wu, Y.Y.; Wang, X.; Li, C.F.; Zhang, F.H. Application of machine learning in bearing fault diagnosis. *Equip. Manuf. Technol.* **2022**, *03*, 03,118–126.

10. Zhang, Y.F.; Li, Y.H.; Wang, D.F.; Kong, L.F. A rolling bearing fault monitoring method based on multi-source information fusion. *Bearing* **2022**, *12*, 12,59–65.

11. Zhang, X.Y.; Luan, Z.Q.; Liu, X.L. A review of deep learning based rolling bearing fault diagnosis research. *Equip. Manag. Maint.* **2017**, *18*, 130–133.

12. Lu, X.Y.; Zhang, C.B.; Gao, J.; Xu, Y.P.; Shao, X. Bearing fault diagnosis algorithm based on convolutional neural network and CatBoost. *Electromechanical Eng.* **2023**, *01*, 1–10.

13. Janssens, O.; Slavkovikj, V.; Vervisch, B.; Stockman, K.; Loccufier, M.; Verstockt, S.; Van de Walle, R.; Van Hoecke, S. Convolutional neural network based fault detection for rotating machinery. *J. Sound Vib.* **2016**, *377*, 331–345. [CrossRef]

14. Gu, X.; Tang, X.H.; Lu, J.G.; Li, S.W. Adaptive Fault Diagnosis Method for Rolling Bearings Based on I-DCNN-LSTM. *Mach. Tool. Hydraul.* **2020**, *48*, 107–113.

15. Zhang, Y.; Liu, Y.; Wang, L.; Li, D.; Zhang, W.; Kong, L. Bearing Non-Uniform Loading Condition Monitoring Based on Dual-Channel Fusion Improved DenseNet Network. *Lubricants* **2023**, *11*, 251. [CrossRef]

16. Jin, X.B.; Lin, Y.S.; Zhang, H. Multisensor fusion estimation in state monitoring. *Control. Theory Appl.* **2009**, *26*, 296–298.

17. Okatan, A.; Hajiyev, C.; Hajiyeva, U. Fault detection in sensor information fusion Kalman filter. *Int. J. Electron. Commun.* **2009**, *63*, 762–768. [CrossRef]

18. Bath, W.G.; Boswell, C.M.; Sommerer, S.; Wang, I. Detection systems information fusion. *Johns Hopkins APL Tech. Dig.* **2005**, *26*, 306–313.

19. Sung, W.T.; Tsai, M.H. Multi-sensor wireless signal aggregation for environmental monitoring system via multi-bit data fusion. *Appl. Math. Inf. Sci.* **2011**, *5*, 589–603.

20. Li, Z.M.; Chen, R.Z.; Zhang, B.M. Study of adaptive weighted estimate algorithm of congeneric multi-sensor data fusion. *J. Lanzhou Univ. Technol.* **2006**, *32*, 78–82.

21. Yan, J.; Hu, Y.; Guo, C. Rotor unbalance fault diagnosis using DBN based on multi-source heterogeneous information fusion. *Procedia Manuf.* **2019**, *35*, 1184–1189. [CrossRef]

22. Duan, Z.S.; Han, C.Z.; Tao, T.F. Consistent multi-sensor data fusion based on nearest statistical distance. *Chin. J. Sci. Instrum.* **2005**, *26*, 478–481.

23. Wang, J.Q.; Zhou, H.Y.; Wu, Y. The theory of data fusion based on state optimal estimation. *Math. Appl.* **2007**, *20*, 392–399.

24. Zheng, Y.J.; Niu, R.X.; Varshney, P.K. Sequential Bayesian estimation with censored data for multi-sensor systems. *IEEE Trans. Signal Process.* **2014**, *62*, 2626–2641. [CrossRef]

25. Fan, S.L.; Li, D.G.; Zhao, J.M. An mine multi-sensor maximum likelihood estimation data fusion algorithm. *J. Inf. Comput. Sci.* **2013**, *10*, 3809–3814. [CrossRef]

26. Lei, X.F.; Zhu, B. Perception of microburst based on multi-sensor data fusion. *Inf. Control.* **2011**, *40*, 296–301.

27. Fincher, D.W.; Mix, D.F. Multi-sensor data fusion using neural networks. In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Los Angeles, CA, USA, 4–7 November 1990; pp. 835–838.

28. Tang, A.H.; Zhang, Y.M. Application of fuzzy clustering in multi-sensor information fusion. *J. Theor. Appl. Inf. Technol.* **2012**, *45*, 661–667.

29. Su, W.X.; Zhu, Y.L.; Liu, F.; Ma, L.B. A homogeneous multi-sensor online data fusion method based on improved fuzzy clustering. *Inf. Control.* **2015**, *44*, 557–563.

30. Tang, Z.Y.; Cai, Y.; Wang, H. A multi-sensor data fusion method based on adaptive weighting algorithm. *Command. Inf. Syst. Technol.* **2022**, *13*, 66–70.

31. Cai, B.L.; Su, G.D. Research on improved batch estimation and adaptive weighted fusion method. *Meas. Control. Technol.* **2019**, *38*, 122–126.

32. Zhu, K.; Song, X.; He, J.X.; Yang, L. Greenhouse data fusion based on wavelet noise reduction and adaptive weighting method. *Jiangsu Agric. Sci.* **2021**, *49*, 180–186.

33. Feng, A.A.; Yue, J.H.; Zheng, Y.; Guo, X.Y. Simulation analysis of wavelet packet denoising optimized by thought evolution algorithm. *Comput. Simul.* **2020**, *37*, 285–290.

34. Guo, C.J.; Gong, C.Y.; Rong, F.; Song, Y.Q. Real-time vibration signal storage management technology based on entropy weight method. *J. Tianjin Polytech. Univ.* **2015**, *34*, 67–71.

35. Chen, L.; Zhang, C.L. Rolling bearing fault diagnosis based on EMD envelope spectral features and PCA-PNN. *Coal Mine Mach.* **2022**, *43*, 173–176.

36. He, X.S.; Yang, X.R. Analytical comparison of particle swarm algorithms for chaotic mapping. *J. Basic Sci. Text. Coll. Univ.* **2023**, *36*, 86–93.

37. Li, K.; Xiong, M.; Su Lei Lu, L.X.; Chen, S. Fault diagnosis method based on improved deep limit learning machine. *Vib. Test Diagn.* **2020**, *40*, 1120–1127.

38. Heo, B.; Chun, S.; Oh, S.J.; Han, D.; Yun, S.; Kim, G.; Uh, Y.; Ha, J. Adamp: Slowing down the slowdown for momentum optimizers on scale-invariant weights. *arXiv* **2020**, arXiv:2006.08217.