energies

MDPI

Article

Fault Diagnosis of Wind Turbine Bearings Based on CEEMDAN-GWO-KELM

Liping Liu [1,2], Ying Wei [1,*], Xiuyun Song [3] and Lei Zhang [4,*]

[1] College of Artificial Intelligence, North China University of Science and Technology, Tangshan 063210, China
[2] College of Mechanical and Energy Engineering, Shanghai Technical Institute of Electronics & Information, Shanghai 201411, China
[3] Faculty of International Languages, Qinggong College, North China University of Science and Technology, Tangshan 064000, China
[4] College of Metallurgy and Energy, North China University of Science and Technology, Tangshan 063009, China
* Correspondence: weiying2717@163.com (Y.W.); zhanglei@ncst.edu.cn (L.Z.)

Abstract: To solve the problem of fault signals of wind turbine bearings being weak, not easy to extract, and difficult to identify, this paper proposes a fault diagnosis method for fan bearings based on Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) and Grey Wolf Algorithm Optimization Kernel Extreme Learning Machine (GWO-KELM). First, eliminating the interference of noise on the collected vibration signal should be conducted, in which the wavelet threshold denoising approach is used in order to reduce the noise interference with the vibration signal. Next, CEEMDAN is used to decompose the signal after a denoising operation to obtain the multi-group intrinsic mode function (IMF), and the feature vector is selected by combining the correlation coefficients to eliminate the spurious feature components. Finally, the fuzzy entropy for the chosen IMF component is input into the GWO-KELM model as a feature vector for defect detection. After diagnosing the Case Western Reserve University (CWRU) dataset by the method presented in this research, it is found that the method can identify 99.42% of the various bearing states. When compared to existing combination approaches, the proposed method is shown to be more efficient for diagnosing wind turbine bearing faults.

Keywords: CEEMDAN; fuzzy entropy; wind turbine; fault diagnosis; bearings; GWO-KELM

Citation: Liu, L.; Wei, Y.; Song, X.; Zhang, L. Fault Diagnosis of Wind Turbine Bearings Based on CEEMDAN-GWO-KELM. *Energies* **2023**, *16*, 48. https://doi.org/10.3390/en16010048

Academic Editors: Tao Yang and Bin Zhang

Received: 17 November 2022
Revised: 13 December 2022
Accepted: 16 December 2022
Published: 21 December 2022

1. Introduction

As a kind of renewable and clean energy source, research on wind energy has developed rapidly in recent years [1]. Wind turbines are a key piece of equipment in wind power generation systems due to their long-term operation in the field in relatively harsh environments, resulting in their frequent failure. As an important part of the fan drive system, the operating condition of the rolling bearings often directly affects the performance of the whole machine [2]. Therefore, it is crucial to monitor and diagnose the rolling bearings of wind turbines in order to improve the safe and stable operation of machinery, lower maintenance costs, and increase the financial advantages of operating a wind farm.

In practical engineering, the fault vibration signal collected by people is nonlinear and non-stationary, and there is noise to a large extent. The method of wavelet threshold denoising [3] can be effective for the noise reduction of vibration signals. Regarding the extraction of fault features, the conventional wind turbine vibration signal processing approach is unable to reliably produce an accurate judgment of wind turbine fault [4]. Empirical Mode Decomposition (EMD) was first proposed by Huang et al. [5], which reduces the algorithm decomposition error caused by human factors and is widely applied to fault diagnosis [6], signal denoising [7], medicine [8], and other fields. Due to the defects of EMD theory, the decomposed signals will result in the modal aliasing phenomenon and end-point effects. To resolve this problem, Wu et al. developed Ensemble Empirical Mode Decomposition (EEMD) [9], an enhanced algorithm of EMD. It adds white Gaussian noise to the original signal to eliminate the mode aliasing phenomenon in the process of EMD decomposition as far as possible; however, there is a lot of noise residual in its reconstructed signal. Torres et al. proposed a new improved algorithm named complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) [10]. This algorithm can well process nonlinear and non-stationary signals, effectively reducing the reconstruction error of EEMD with high decomposition efficiency and good decomposition effect. Colominas et al. [11] have improved CEEMDAN and successfully tested it on several biomedical signals. Wang and Shao [12] used reweighted CEEMDAN denoising to extract fault features. Hassan et al. [13] used CEEMDAN decomposition to identify seizures.

One of the crucial processes in mechanical problem diagnostics is feature extraction. In recent years, dimensionless parameters have been added to the feature extraction of mechanical equipment vibration signals, including approximate entropy, sample entropy, and fuzzy entropy. Sampaio et al. [14] used the approximate entropy algorithm to the vibration signal of the crack axis to detect the crack and its depth. Wang et al. [15] accurately classified the fault samples of rolling bearings using the generalized fine composite multiscale sample entropy algorithm. Effective feature selection for diagnosing chattering vibrations was achieved by Tran et al. [16] using fuzzy entropy and a similarity classifier. After feature extraction, a suitable classification algorithm is needed to implement fault diagnosis. The single-layer feed-forward neural network Extreme Learning Machine (ELM) [17] has good learning ability and generalization ability, so its application field is very wide. Yang et al. [18] used the ELM algorithm to predict the degree of roasting of cocoa beans. Li et al. [19] established a model based on MPA-ELM to predict the thermal displacement of the electric spindle. Katılmış et al. [20] used an ELM classifier in a sign language recognition system. Diker et al. [21] improved the classification performance of ECG signals using the DEA-ELM method. He et al. [22] used the reverse cognition drosophila optimization algorithm to optimize ELM parameters and established a rolling bearing fault diagnosis model. However, ELM makes the classification results fluctuate and the stability is relatively poor. For this reason, Huang et al. [23] introduced kernel mapping in ELM instead of random mapping, which enhanced its stability and generalization ability. Hu et al. [24] used the particle swarm algorithm to optimize KELM for planetary gearbox problem detection in wind turbines and achieved certain results.

To sum up, this research suggests a method for detecting wind turbine bearing problems based on CEEMDAN-GWO-KELM, aiming at the issue that it is challenging to accurately extract various operating states of wind turbine bearings in an environment with a lot of background noise. Firstly, use the wavelet threshold denoising method to denoise the bearing defect signal, then the denoised signal is subjected to CEEMDAN decomposition to produce a series of IMF components. The first four IMFs with rich bearing characteristic information are selected by the correlation coefficient, and the fuzzy entropy of the first four IMFs is calculated. Finally, to diagnose wind turbine bearing faults by inputting the fuzzy entropy as a feature vector into the GWO-KELM model. This method has proven efficient for diagnosing wind turbine bearing faults when compared with other combination methods.

2. Fundamental

2.1. Wavelet Threshold Denoising

Wavelet threshold denoising is the decomposition of the signal containing noise at each scale by wavelet, and all the decomposition values at a large scale (low resolution) are retained. Both the soft threshold method and the hard threshold method can be applied to cope with small-scale (high-resolution) decomposition values. Finally, the wavelet transform is used for reconstruction, and the effective signal is recovered. The basic principle is shown in Figure 1. Wavelet threshold denoising functions usually include both soft thresholding and hard thresholding. The methods of soft threshold and hard threshold denoising differ. The generated signal is as smooth as the original signal because of the excellent continuity of the soft threshold denoising method; however, the hard threshold denoising method lacks smoothness. Therefore, the soft threshold denoising method is used in this research. For soft threshold expression, see Formula (1):

$$\omega\lambda = \begin{cases} [sign(\omega)](|\omega| - \lambda), & |\omega| \geq \lambda \\ 0, & |\omega| < \lambda \end{cases} \tag{1}$$

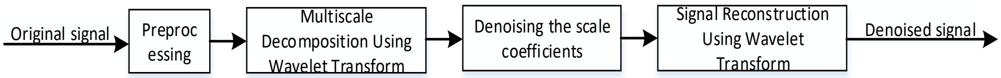

Original signal
Preproc essing
Multiscale Decomposition Using Wavelet Transform
Denoising the scale coefficients
Signal Reconstruction Using Wavelet Transform
Denoised signal

Figure 1. Schematic diagram of wavelet threshold denoising.

Among them, $sign(\cdot)$ is a symbolic function; ω is the magnitude of the wavelet coefficients; $\omega\lambda$ is the magnitude of wavelet coefficients after adding the threshold; and λ is the threshold.

In this study, the threshold value needs to be obtained by using the *wden* function in MATLAB. Here, *wden* uses a function in the wavelet toolbox of MATLAB R2019a. The wavelet coefficients need to be obtained by the *wden* function using the specified orthogonal or double orthogonal wavelets to perform N-level wavelet decomposition of the signal. For the *wden* function, see Formula (2):

$$xd = wden(x, tptr, sorh, scal, n, wname) \tag{2}$$

In the formula, *xd* is a denoised signal by wavelet thresholding; x is the original signal; *tptr* is the threshold selection criterion; *sorh* is the threshold selection method; n is the number of layers to be decomposed; and *wname* is the wavelet basis function.

2.2. CEEMDAN Algorithm

Based on the foundation of EMD, CEEMDAN also takes the concept of averaging by multiple superpositions while incorporating white Gaussian noise into the original signal. Here are the steps to break it down:

Step 1: Construct the signal $X(t)$, as shown in Formula (3):

$$X(t) = x(t) + \varepsilon_0 \omega_i(t) \tag{3}$$

In the formula, $x(t)$ is the original signal; $X(t)$ is the signal after adding white noise; ε_0 is the amplitude; and $\omega_i(t)(i = 1, 2, \ldots, N)$ is white noise.

Step 2: The signal *X(t)* was decomposed by EMD to obtain the first-order modal component, as shown in Formula (4):

$$IMF_1(n) = \frac{1}{N}\sum_{i=1}^{N} IMF_1^i(n) \tag{4}$$

Step 3: After the first-order modal component is obtained, the first residual component is calculated, as shown in Formula (5):

$$r_1(n) = x(n) - IMF_1(n) \tag{5}$$

Step 4: Through the obtained $r_1(n)$, decompose again to obtain the second-order modal component, as shown in Formula (6):

$$IMF_2(n) = \frac{1}{N}\sum_{i=1}^{N} E_1\left\{r_1(n) + \varepsilon_1 E_1\left[\omega^i(n)\right]\right\} \tag{6}$$

Step 5: Calculate the jth order residual component from $j = 2, 3, \cdots$ to obtain the $j + 1$th order modal component, then the jth residual component and the modal component, as shown in Formulas (7) and (8):

$$\mathrm{r}_j(n) = r_{j-1}(n) - IMF_n(n) \tag{7}$$

$$IMF_j(n) = \frac{1}{N}\sum_{i=1}^{N} E_1\left\{r_j(n) + \varepsilon_j E_j\left[\omega^i(n)\right]\right\} \tag{8}$$

In the formula, $E_j(\cdot)$ is the jth-order modal component.

Step 6: Repeat the above steps until there are no more than two extreme points of the residual signal. When the algorithm is terminated, the j-order modal component can be obtained, and the final residual signal is shown in Formula (9):

$$r(n) = x(n) - \sum_{j=1}^{J} IMF_j \tag{9}$$

Finally, the original signal after decomposition is shown in Formula (10):

$$x(n) = \sum_{j=1}^{J} IMF_j + r(n) \tag{10}$$

2.3. Fuzzy Entropy

Fuzzy Entropy is improved on the basis of sample entropy and is a quantitative statistical indicator of signal complexity. It is different from sample entropy in that sample entropy uses a unit step function, which has a large mutation and lacks the continuity of an entropy value. The value of the threshold is sensitive to it, and its weak change may lead to the mutation of the sample entropy value. Fuzzy entropy combines fuzzy set theory and uses a fuzzy membership function as the hard threshold criterion in entropy. This paper constructs the feature vector based on the fuzzy entropy of the IMF component after Pearson correlation coefficient selection to characterize the state information contained in the signal. The specific calculation process of fuzzy entropy is as follows:

Step 1: With respect to a time series of length N $\{x(i), i = 1, 2, \ldots, N\}$, initialize the embedding dimension m. The above time series is reconstructed in phase space, as shown in Formula (11):

$$X(i) = \{x(i), x(i+1), \ldots, x(i+m-1)\} - u(i) \tag{11}$$

In the formula, $X(i)$ is the new time series after reconstruction, $i = 1, 2, \ldots, N - m + 1$, and $u(i)$ is the average value of m consecutive $x(i)$, as shown in Formula (12):

$$u_i = \frac{1}{m}\sum_{k=0}^{m-1} x(i+k) \tag{12}$$

Step 2: Definition of the distance between two vectors $X(i)$ and $X(j)$ for the difference in the corresponding element of absolute value maximum, as shown in Formula (13):

$$d_{ij}^m = \max\{|(x(i+k) - u(i)) - (x(j+k) - u(i))|\} \tag{13}$$

In the formula, $1 \le i, j \le N - m + 1$, and $i \ne j$.

Step 3: To define the similarity, introduce a fuzzy membership function between vectors $X(i)$ and $X(j)$, as shown in Formula (14):

$$A_{ij}^m \begin{cases} 1, & d_{ij}^m = 0 \\ \exp\left[-\ln(2)\left(\frac{d_{ij}^m}{r}\right)^2\right], & d_{ij}^m > 0 \end{cases} \tag{14}$$

In the formula, r is the similarity tolerance parameter, δ is the standard deviation of the original 1D time series, define the relationship between the two as: $r = R \times \delta$.

Step 4: Define the function $C_i^m(r)$, as shown in Formula (15):

$$C_i^m(r) = \frac{1}{N-m} \sum_{j=1, j \neq i}^{N-m+1} A_{ij}^m \tag{15}$$

Then, the relation dimension under m dimension can be obtained, as shown in Formula (16):

$$\Phi^m(r) = \frac{1}{N-m} \sum_{i=1}^{N-m+1} C_i^m(r) \tag{16}$$

Step 5: Increase the embedding dimension by 1, and then repeat the above steps 1 to 4 for the $m+1$ dimension vector to obtain the relational dimension $\Phi^{m+1}(r)$ in the $m+1$ dimension, as shown in Formula (17):

$$\Phi^{m+1}(r) = \frac{1}{N-m} \sum_{i=1}^{N-m} C_i^{m+1}(r) \tag{17}$$

Step 6: Finally, the expression of fuzzy entropy can be obtained as:

$$FuzzyEn(m, r, N) = \ln \Phi^m(r) - \ln \Phi^{m+1}(r) \tag{18}$$

In the formula, m is the embedding dimension parameter, and N is the original time series length.

3. Fault Diagnosis Based on GWO-KELM

3.1. Grey Wolf Optimization Algorithm

The grey wolf optimization algorithm is a new intelligent optimization algorithm for simulating the social hierarchy and hunting behavior of wolf packs. This algorithm has strong convergence and fewer parameters. The flow chart of the GWO algorithm is shown in Figure 2. There are four social classes of wolves in GWO, in which the α wolves have the highest rank and lead the other wolves; the second is the β wolves, which help the α wolves make decisions; and the δ wolves follow the first two wolves and command the ω wolves; finally, the ω wolves, who are subordinate to the α, β, and δ wolves. The first three types of wolves hunt down their prey, and the last wolves track and encircle them to achieve the purpose of catching their prey [25]. The process of the GWO algorithm is as follows:

Step 1: Surround the prey. In pursuit of their prey, the wolves surround it first. The mathematical modeling of this behavior is:

$$\begin{cases} \boldsymbol{D} = |\boldsymbol{C} \cdot \boldsymbol{X}_P(t) - \boldsymbol{X}(t)| \\ \boldsymbol{X}(t+1) = \boldsymbol{X}_P(t) - \boldsymbol{A} \cdot D \end{cases} \tag{19}$$

$$\begin{cases} \boldsymbol{A} = 2\boldsymbol{A}\boldsymbol{r}_1 - \boldsymbol{A} \\ \boldsymbol{C} = 2\boldsymbol{r}_2 \end{cases} \tag{20}$$

In the formula, $\boldsymbol{D}$ is the distance between grey wolf and prey; $\boldsymbol{X}_P(t)$ is the current position of the prey; t is the number of iterations; $\boldsymbol{X}(t)$ is the current position of the grey wolf; $\boldsymbol{r}_1, \boldsymbol{r}_2 \in [0, 1]$; α is the convergence factor; and $\boldsymbol{A}$ and $\boldsymbol{C}$ are the coefficient vectors.

Step 2: Hunt down the prey. When the grey wolf identifies the location of the prey, α leads β and δ to guide the wolf pack to surround the prey. Mathematically modeled as:

$$\begin{cases} \boldsymbol{D}_\alpha = |\boldsymbol{C}_1 \cdot \boldsymbol{X}_\alpha - \boldsymbol{X}| \\ \boldsymbol{D}_\beta = |\boldsymbol{C}_2 \cdot \boldsymbol{X}_\beta - \boldsymbol{X}| \\ \boldsymbol{D}_\delta = |\boldsymbol{C}_3 \cdot \boldsymbol{X}_\delta - \boldsymbol{X}| \end{cases} \tag{21}$$

In the formula, $\boldsymbol{X}$ is the current wolf position; $\boldsymbol{C}_1$, $\boldsymbol{C}_2$, and $\boldsymbol{C}_3$ correspond to coefficient vectors of α, β, and δ; $\boldsymbol{D}_\alpha$, $\boldsymbol{D}_\beta$, and $\boldsymbol{D}_\delta$ represent the distances between α, β, and δ and the prey, respectively; and $\boldsymbol{X}_\alpha$, $\boldsymbol{X}_\beta$, and $\boldsymbol{X}_\delta$ represent the location of the corresponding wolf pack at the moment.

Step 3: Update individual grey wolf.

$$\begin{cases} \boldsymbol{X}_1 = \boldsymbol{X}_\alpha - \boldsymbol{A}_1 \cdot \boldsymbol{D}_\alpha \\ \boldsymbol{X}_2 = \boldsymbol{X}_\beta - \boldsymbol{A}_2 \cdot \boldsymbol{D}_\beta \\ \boldsymbol{X}_3 = \boldsymbol{X}_\delta - \boldsymbol{A}_3 \cdot \boldsymbol{D}_\delta \end{cases} \tag{22}$$

$$\boldsymbol{X}(t+1) = \frac{\boldsymbol{X}_1 + \boldsymbol{X}_2 + \boldsymbol{X}_3}{3} \tag{23}$$

In the formula, $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, and $\boldsymbol{A}_3$ correspond to coefficient vectors of α, β, and δ; $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, and $\boldsymbol{X}_3$ represent the direction vectors that α, β, and δ guide ω to move in the next step; and $\boldsymbol{X}(t+1)$ represents the place of the candidate wolf that the three wolf packs produced.

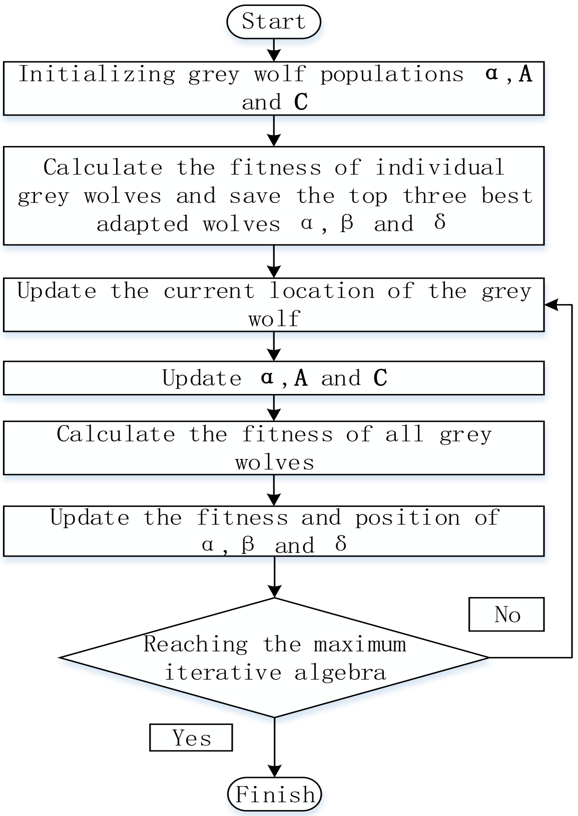

Start
Initializing grey wolf populations α, A and C
Calculate the fitness of individual grey wolves and save the top three best adapted wolves α, β and δ
Update the current location of the grey wolf
Update α, A and C
Calculate the fitness of all grey wolves
Update the fitness and position of α, β and δ
Reaching the maximum iterative algebra
No
Yes
Finish

Figure 2. GWO algorithm flow chart.

3.2. Kernel Extreme Learning Machine (KELM)

Huang and others proposed ELM for the first time in 2006. It performs generalization well and learns incredibly and quickly. The difference between KELM and the ELM algorithm is that a kernel function is added on the basis of achieving the purpose of enhancing the stability and generalization of the ELM algorithm. The traditional ELM algorithm expression is:

$$\varphi(x) = h(x)\beta \tag{24}$$

$$H = h(x) = \begin{bmatrix} g(\omega_1 x_1 + b_1) & \cdots & g(\omega_k x_1 + b_k) \\ \vdots & & \vdots \\ g(\omega_1 x_N + b_1) & \cdots & g(\omega_k x_N + b_k) \end{bmatrix}_{N \times k} \tag{25}$$

$$\beta = H^{+}T \tag{26}$$

$$H^{+} = H^{T}\left(HH^{T}\right)^{-1} \tag{27}$$

In the formula, N is the number of samples; ω is the weight between input and output; b is the bias of the hidden layer; H is the output matrix of the hidden layer; H^{+} is the generalized inverse matrix of H; β is the vector of weights between the output and the hidden layer; $g()$ is activation function; k is the quantity of concealed units; and T is the target matrix for the training set.

KELM makes the kernel mapping in the model more stable by introducing kernel functions. The expression of the KELM algorithm is:

$$\varphi(x) = h(x)H^{T}\left(\frac{I}{C} + HH^{T}\right)^{-1}T = \begin{bmatrix} \mathrm{k}(x, x_1) \\ \vdots \\ \mathrm{k}(x, x_N) \end{bmatrix}\left(\frac{I}{C} + HH^{T}\right)^{-1}T \tag{28}$$

In the formula, C is the regularization parameter; I is an N order unit matrix; and $k(A, b)$ is the kernel function; here, the RBF kernel function is adopted and is needed to set the kernel parameter γ.

3.3. GWO Optimizes the KELM Model

In this study, the Kernel Extreme Learning Machine is optimized using the grey wolf method. In the GWO-KELM model, GWO is used to optimize the parameters C and γ of KELM. The radial basis function is chosen as the KELM kernel function in this study. Figure 3 depicts the GWO-KELM process, and the specific operational process is described as follows:

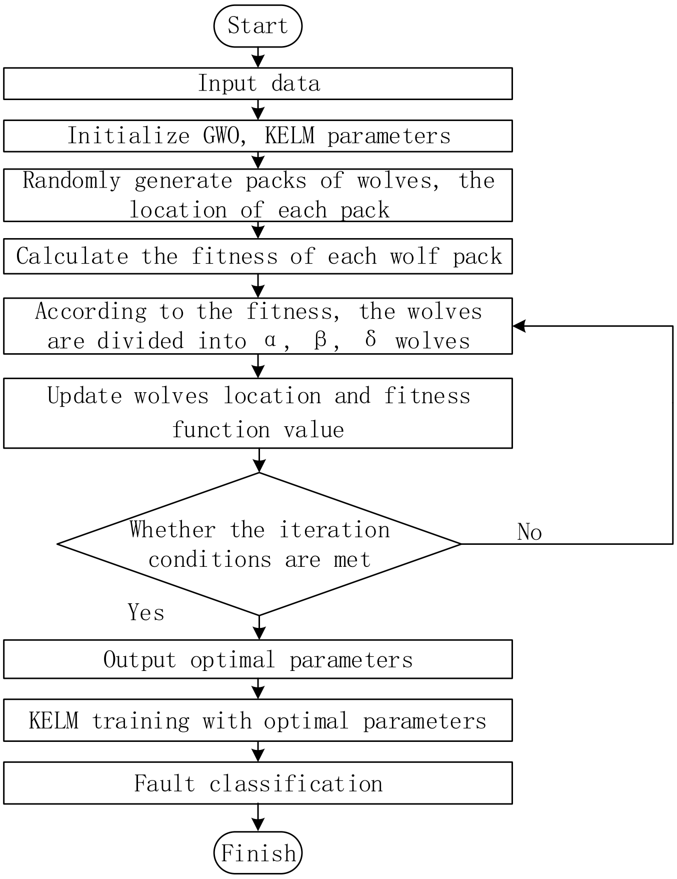

Start
Input data
Initialize GWO, KELM parameters
Randomly generate packs of wolves, the location of each pack
Calculate the fitness of each wolf pack
According to the fitness, the wolves are divided into α, β, δ wolves
Update wolves location and fitness function value
Whether the iteration conditions are met
No
Yes
Output optimal parameters
KELM training with optimal parameters
Fault classification
Finish

Figure 3. Flow chart of the proposed GWO-KELM method.

Step 1: Input data.

Step 2: GWO and KELM parameters are initialized, including population size, iteration times, regularization coefficient, and kernel parameters, and KELM recognition accuracy is taken as the fitness function.

Step 3: Randomly generate Wolf pack positions within the set range.

Step 4: Determine the value of the wolf pack's fitness function at its current location.

Step 5: Wolves are divided into α, β, and δ groups according to their fitness.

Step 6: Update the wolf pack location and fitness function values.

Step 7: Check to see if the desired number of iterations is reached, if not, return to step 5; otherwise, the optimization ends, and the output is the optimal parameters C and γ.

Step 8: Train KELM with optimal parameters for fault classification.

3.4. Fault Diagnosis Model

Based on CEEMDAN-GWO-KELM wind turbine bearing fault diagnosis method is depicted in Figure 4 and involves the following steps:

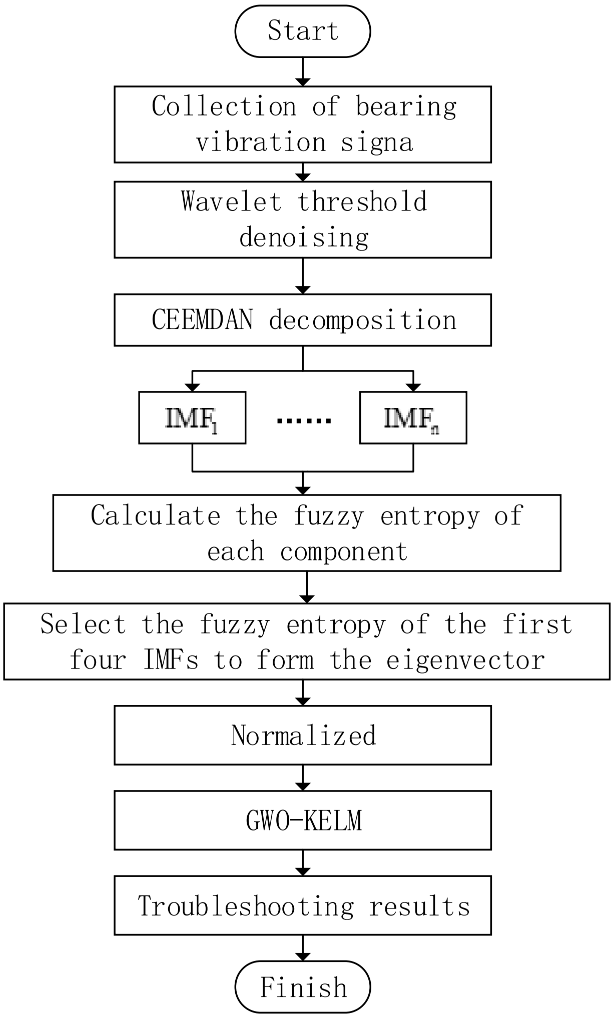

Start
Collection of bearing vibration signa
Wavelet threshold denoising
CEEMDAN decomposition
IMF_1
……
IMF_n
Calculate the fuzzy entropy of each component
Select the fuzzy entropy of the first four IMFs to form the eigenvector
Normalized
GWO-KELM
Troubleshooting results
Finish

Figure 4. The overall fault diagnosis flowchart.

Step 1: The acquisition of vibration signals, the acceleration signal data of various bearing states are gathered at a specific sample frequency.

Step 2: In data preprocessing, first use the wavelet threshold denoising method for noise reduction processing, then decompose the denoised signal by the method of CEEMDAN and obtain some IMF components.

Step 3: Feature extraction, calculate fuzzy entropy of each IMF component. The fuzzy entropy of the first four-order IMF components is selected by the correlation coefficient, and constructs the feature vector.

Step 4: Normalization processing, normalizing the extracted fault feature dataset.

Step 5: In fault diagnosis, use the processed feature dataset as the GWO-KELM fault classifier's input to achieve fault diagnosis.

4. Experimental Data Processing and Analysis

4.1. Data Collection and Preprocessing

This article used Case Western Reserve University (CWRU) bearing simulation experiment data centers to evaluate the accuracy and efficacy of CEEMDAN-GWO-KELM methods in wind turbine bearing defect diagnostics. A driver, a torque sensor, a load click, and a test bearing make up the test bench, as shown in Figure 5.

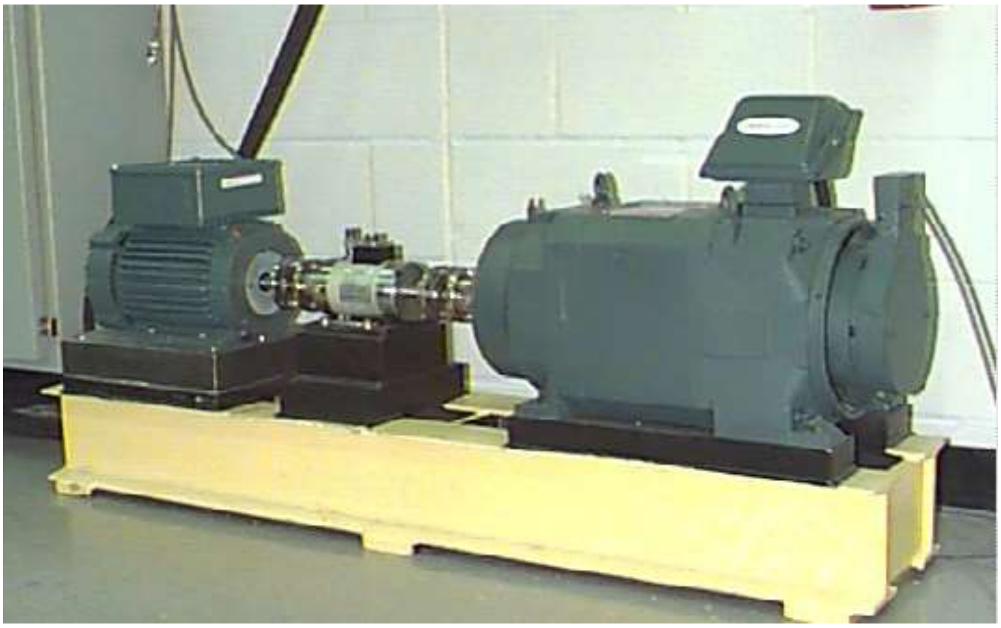


Figure 5. Rolling bearing fault test rig.

In the bearing experiment of CWRU, the motor shaft is supported by the bearing that will be tested. The drive end bearing model number SKF6205 has sampling rates of 12 and 48 kHz, respectively; SKF6203 is the fan end bearing, while 12 KHz is the sampling frequency. Its working principle is similar to that of the wind turbine drive chain, and the rotational speeds of the bearings in the CWRU dataset are 1730 rpm, 1750 rpm, 1772 rpm, and 1797 rpm, which are within the rotational speed range of the wind turbine bearings. The driving characteristics and the CWRU bearing test bench's rotating speed are similar to those of the wind turbine bearing; consequently, this dataset is used to verify the algorithm proposed in this research.

In order to confirm the algorithm's efficacy in fault diagnosis, the normal operating data and damage diameters of the bearing at a rotational speed of 1730 rpm and a sampling frequency of 12 KHz are selected as 0.1778 mm (0.007 inches), 0.3556 (0.014 inches), and 0.5334 (0.021 inches) operating data for inner ring failure, outer ring failure, and rolling element failure. The normal bearings are grouped into one group and the remaining faulty bearings are grouped into nine groups. As a result, ten labels in total are defined, as seen in Table 1. The original data sample length is divided into 2048 points, each state signal has 30 groups, and the length of each group of signals is 2048.

In order to understand the process more intuitively, the bearing example of the inner ring fault is used to discuss. Figure 6 displays the experimental signal and the inner ring fault's frequency spectrum. Although the signal clearly exhibits shock characteristics, it cannot directly determine the type of fault, so a further process is needed.

Table 1. Fault type and fault code.

Fault Type	Fault Size (mm)	Fault Code
Inner ring fault 1	0.1778	1
Inner ring fault 2	0.3556	2
Inner ring fault 3	0.5334	3
Outer ring fault 1	0.1778	4
Outer ring fault 2	0.3556	5
Outer ring fault 3	0.5334	6
Rolling element failure 1	0.1778	7
Rolling element failure 2	0.3556	8
Rolling element failure 3	0.5334	9
normal	0	10

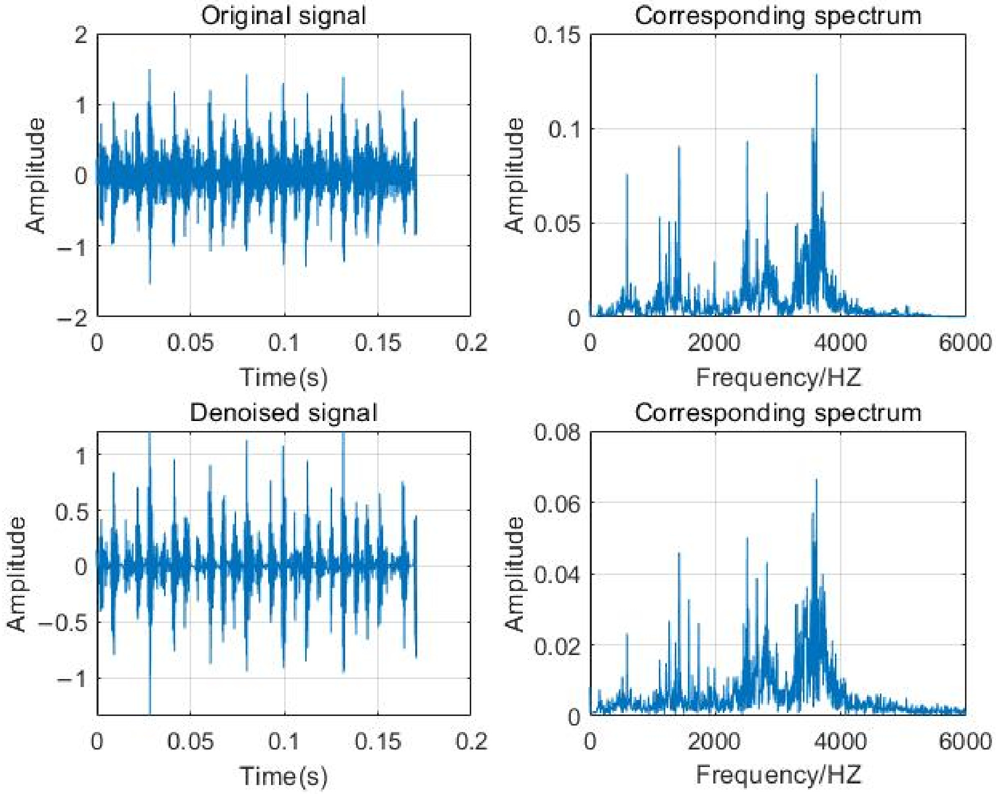


Figure 6. Waveform and spectrum of inner circle fault signal.

4.2. Feature Extraction Based on CEEMDAN Decomposition

The inner ring fault signal after noise removal is broken down by CEEMDAN, and decomposition yields a number of IMF components. The larger the order, the lower the frequency; that is, the less fault information it contains.

In order to accurately extract the IMF components with more fault signals, the correlation coefficient is selected to screen each modal component. The correlation coefficient of each IMF is calculated. In order to further screen out the appropriate components, the correlation coefficient of the $IMFs$ of ten different fault signals is calculated. The first seven IMF correlation coefficients of the ten fault types are shown in Table 2.

Table 2. *IMF* correlation coefficient of ten fault types.

Fault Type	IMF_1	IMF_2	IMF_3	IMF_4	IMF_5	IMF_6	IMF_7	Fault Code
Inner ring fault 1	0.914429	0.382676	0.254300	0.144165	0.102502	0.047260	0.029596	1
Inner ring fault 2	0.907739	0.428273	0.318125	0.125363	0.108809	0.074792	0.044658	2
Inner ring fault 3	0.987412	0.141102	0.100381	0.079551	0.082432	0.051613	0.023653	3
Outer ring fault 1	0.978397	0.260789	0.039529	0.006446	0.017525	0.017619	0.007012	4
Outer ring fault 2	0.398757	0.159746	0.336387	0.521029	0.560815	0.627021	0.413391	5
Outer ring fault 3	0.965937	0.150558	0.145045	0.168494	0.068543	0.021311	0.013905	6
Rolling element failure 1	0.348466	0.224356	0.160948	0.379321	0.619729	0.72292	0.303485	7
Rolling element failure 2	0.752964	0.379008	0.422717	0.400106	0.468957	0.166869	0.085792	8
Rolling element failure 3	0.714964	0.463578	0.528637	0.379162	0.357800	0.339578	0.151980	9
normal	0.468767	0.530276	0.596383	0.515570	0.430901	0.506312	0.439301	10

The table above demonstrates that the correlation coefficients between the *IMF* components of the first four faults and the original signals are large, most of which are more than 0.1, while the correlation coefficients of the latter are mostly lower than 0.1, or even tend to 0. From this, it can be concluded that the first four *IMF* components have the largest correlation with the original signal and contain the primary fault information. Therefore, the fuzzy entropy of the first four *IMF* components is determined as the feature vector, and the remainder is employed as a false component to be eliminated.

Fuzzy entropy can characterize fault characteristics to a certain extent, and the fuzzy entropy characteristic values under 10 states are as displayed in Table 3.

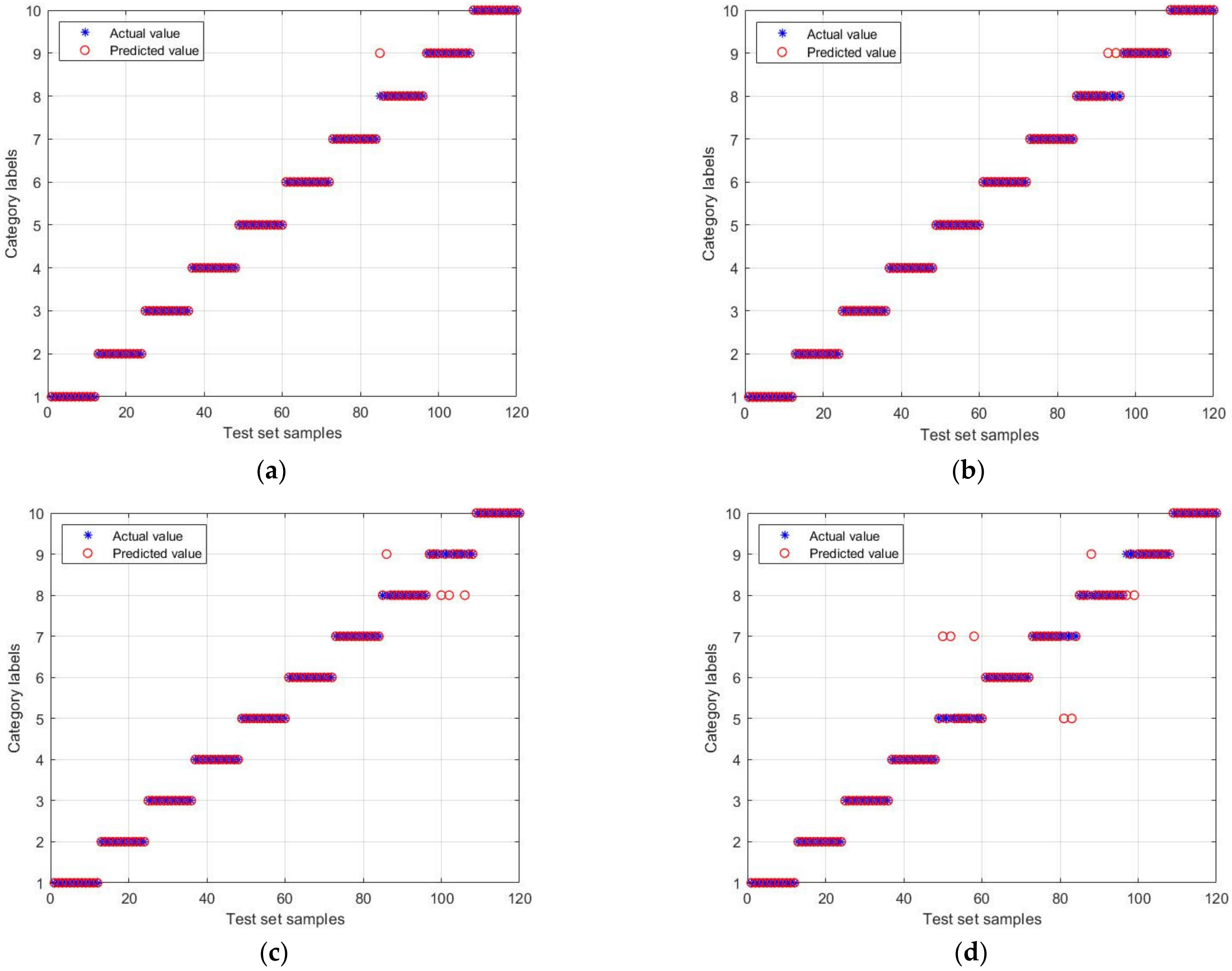
Table 3. *IMF* Component Fuzzy Entropy of CEEMDAN Decomposition.

Fault Type	IMF_1	IMF_2	IMF_3	IMF_4	Fault Code
Inner ring fault 1	0.434972	0.177999	0.113561	0.031011	1
Inner ring fault 2	0.195311	0.096538	0.113142	0.017670	2
Inner ring fault 3	0.448848	0.114559	0.07165	0.022327	3
Outer ring fault 1	0.276833	0.172664	0.099691	0.02737	4
Outer ring fault 2	0.043899	0.028376	0.008482	0.009212	5
Outer ring fault 3	0.234330	0.093997	0.089899	0.062964	6
Rolling element failure 1	0.028447	0.006922	0.007336	0.002261	7
Rolling element failure 2	0.265729	0.05313	0.076417	0.015427	8
Rolling element failure 3	0.115616	0.039859	0.038023	0.009757	9
normal	0.198319	0.018702	0.068489	0.037776	10

4.3. Experimental Comparative Analysis

In order to confirm the superiority and precision of the GWO-KELM diagnostic model proposed in this paper, the GWO-KELM model is compared with the other three models including the KELM model, GOA-KELM model, and WOA-KELM, to verify the efficiency of each method's classification.

The fuzzy entropy values of the first four IMF components of each fault type constitute a 300 × 4 fault feature matrix and input it into the GWO-KELM model. Eighteen groups of feature vectors in each state were picked at random to serve as training samples, and the remaining 12 groups of feature vectors were selected as test samples. The number of wolves in the GWO-KELM model was set as 20, and 30 iterations was the set maximum number of iterations. Figure 7a presents the diagnostic findings. To demonstrate the classification impact of the model, the WOA-KELM, GOA-KELM, and KELM models were tested with this test set, and Figure 7b–d displays the diagnostic outcomes.



(**a**) (**b**) (**c**) (**d**)

Figure 7. (**a**) GWO-KELM model; (**b**) WOA-KELM model; (**c**) GOA-KELM model; (**d**) KELM model.

Figure 7 shows that the GWO-KELM model suggested in this paper has a classification accuracy of 99.17%, while the classification accuracy of the WOA-KELM, GOA-KELM, and KELM models is 98.33%, 96.67%, and 93.33%, respectively. A single test cannot accurately depict the model's classification accuracy. For the validity and veracity of the experimental findings, the experiments were repeated 20 times on the four models, respectively, and the average value of classification accuracy was used as the foundation. The obtained diagnostic results obtained are depicted in Table 4.

Table 4. Comparison of results of different models.

Diagnostic Model	Average Training Accuracy	Average Test Accuracy
KELM	93.75%	92.29%
GOA-KELM	98.28%	97.06%
WOA-KELM	100%	98.73%
GWO-KELM	100%	99.42%

As seen in Table 4, the average training accuracy of the GWO-KELM model used in this paper is 100%, and the average test accuracy is 99.42%, which is better than the KELM, GOA-KELM, and WOA-KELM models. Therefore, the method proposed in this paper is superior to other diagnostic methods and has certain superiority in rolling bearing fault diagnosis.

5. Conclusions

In summary, aiming at the issue of rolling bearing malfunction identification and diagnosis, the bearing defect diagnosis approach proposed in this paper combines wavelet threshold denoising, CEEMDAN fuzzy entropy, and GWO-KELM. Based on a comparative experiment, this method's efficacy and accuracy are confirmed, which has certain advantages in wind turbine fault diagnosis.

The wavelet threshold denoising approach is used to pretreat the bearing vibration signal, which can reduce noise interference with the vibration signal.

The application of CEEMDAN decomposition to fan-bearing fault diagnosis has greatly reduced the mode aliasing problem. The fuzzy entropy of effective components selected by the Pearson correlation coefficient can better reflect the fault characteristic information of bearings.

By diagnosing various bearing states using the measured bearing data from the CWRU bearing test bench, the validity of the method provided in this work is demonstrated. The experimental comparison shows that the method has higher diagnostic accuracy.

Author Contributions: Conceptualization, Y.W. and L.L.; methodology, L.L. and Y.W.; software, Y.W. and L.L.; validation, Y.W. and L.L.; investigation, Y.W.; resources, L.L.; data curation, Y.W.; writing—original draft preparation, Y.W.; writing—review and editing, Y.W., L.L., L.Z. and X.S.; supervision, L.L., L.Z. and X.S.; project administration, Y.W., L.L., L.Z. and X.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the Hebei Provincial Science and Technology Plan under grant 20327218D.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This study uses the dataset from CWRU. This data can be found here: https://engineering.case.edu/bearingdatacenter/download-data-file (accessed on 1 November 2022).

Acknowledgments: The authors gratefully appreciate the anonymous Reviewers for their valuable comments.

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

Abbreviations

Abbreviations	Full name
CEEMDAN	Complete Ensemble Empirical Mode Decomposition with Adaptive Noise
GWO	Grey Wolf Optimizer
KELM	Kernel Extreme Learning Machine
IMF	Intrinsic Mode Function
CWRU	Case Western Reserve University
EMD	Empirical Mode Decomposition
EEMD	Ensemble Empirical Mode Decomposition
ELM	Extreme Learning Machine
MPA	Marine Predator Algorithm
ECG	Electrocardiogram
DEA	Differential Evolution Algorithm
RBF	Radial Basis Function
GOA	Grasshopper Optimization Algorithm
WOA	Whale Optimization Algorithm

References

1. Artigao, E.; Martin-Martinez, S.; Honrubia-Escribano, A.; Gomez-Lazaro, E. Wind Turbine Reliability: A Comprehensive Review towards Effective Condition Monitoring Development. *Appl. Energy* **2018**, *228*, 1569–1583. [CrossRef]
2. Chen, J.; Pan, J.; Li, Z.; Zi, Y.; Chen, X. Generator Bearing Fault Diagnosis for Wind Turbine via Empirical Wavelet Transform Using Measured Vibration Signals. *Renew. Energy* **2016**, *89*, 80–92. [CrossRef]

3. Wang, F.; Cen, J.; Yu, Z.; Deng, S.; Zhang, G. Research on a Hybrid Model for Cooling Load Prediction Based on Wavelet Threshold Denoising and Deep Learning: A Study in China. *Energy Rep.* **2022**, *8*, 10950–10962. [CrossRef]
4. Liu, W.Y.; Tang, B.P.; Han, J.G.; Lu, X.N.; Hu, N.N.; He, Z.Z. The Structure Healthy Condition Monitoring and Fault Diagnosis Methods in Wind Turbines: A Review. *Renew. Sustain. Energy Rev.* **2015**, *44*, 466–472. [CrossRef]
5. Huang, N.; Shen, Z.; Long, S.; Wu, M.; Shih, H.; Zheng, Q.; Yen, N.; Tung, C.; Liu, H. The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis. *Proc. R. Soc. A-Math. Phys. Eng. Sci.* **1998**, *454*, 903–995. [CrossRef]
6. Mejia-Barron, A.; Valtierra-Rodriguez, M.; Granados-Lieberman, D.; Olivares-Galvan, J.C.; Escarela-Perez, R. The Application of EMD-Based Methods for Diagnosis of Winding Faults in a Transformer Using Transient and Steady State Currents. *Measurement* **2018**, *117*, 371–379. [CrossRef]
7. Ma, X.; Hu, J.; Zhang, L. EMD-Based Online Filtering of Process Data. *Control Eng. Pract.* **2017**, *62*, 79–91. [CrossRef]
8. Sukriti; Chakraborty, M.; Mitra, D. A Novel Automated Seizure Detection System from EMD-MSPCA Denoised EEG: Refined Composite Multiscale Sample, Fuzzy and Permutation Entropies Based Scheme. *Biomed. Signal Process. Control* **2021**, *67*, 102514. [CrossRef]
9. Wu, Z.; Huang, N.E. Ensemble Empirical Mode Decomposition: A Noise-Assisted Data Analysis Method. *Adv. Data Sci. Adapt. Anal.* **2009**, *1*, 1–41. [CrossRef]
10. Torres, M.E.; Colominas, M.A.; Schlotthauer, G.; Flandrin, P. A Complete Ensemble Empirical Mode Decomposition with Adaptive Noise. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011; pp. 4144–4147.
11. Colominas, M.A.; Schlotthauer, G.; Torres, M.E. Improved Complete Ensemble EMD: A Suitable Tool for Biomedical Signal Processing. *Biomed. Signal Process. Control* **2014**, *14*, 19–29. [CrossRef]
12. Wang, L.; Shao, Y. Fault Feature Extraction of Rotating Machinery Using a Reweighted Complete Ensemble Empirical Mode Decomposition with Adaptive Noise and Demodulation Analysis. *Mech. Syst. Signal Process.* **2020**, *138*, 106545. [CrossRef]
13. Hassan, A.R.; Subasi, A.; Zhang, Y. Epilepsy Seizure Detection Using Complete Ensemble Empirical Mode Decomposition with Adaptive Noise. *Knowl.-Based Syst.* **2019**, *191*, 105333. [CrossRef]
14. Sampaio, D.L.; Nicoletti, R. Detection of Cracks in Shafts with the Approximated Entropy Algorithm. *Mech. Syst. Signal Process.* **2016**, *72–73*, 286–302. [CrossRef]
15. Wang, Z.; Yao, L.; Cai, Y. Rolling Bearing Fault Diagnosis Using Generalized Refined Composite Multiscale Sample Entropy and Optimized Support Vector Machine. *Measurement* **2020**, *156*, 107574. [CrossRef]
16. Tran, M.-Q.; Elsisi, M.; Liu, M.-K. Effective Feature Selection with Fuzzy Entropy and Similarity Classifier for Chatter Vibration Diagnosis. *Measurement* **2021**, *184*, 109962. [CrossRef]
17. Huang, G.-B.; Zhu, Q.-Y.; Siew, C.-K. Extreme Learning Machine: Theory and Applications. *Neurocomputing* **2006**, *70*, 489–501. [CrossRef]
18. Yang, Y.; Darwish, A.G.; El-Sharkawy, I.; Zhu, Q.; Sun, S.; Tan, J. Rapid Determination of the Roasting Degree of Cocoa Beans by Extreme Learning Machine (ELM)-Based Imaging Analysis. *J. Agric. Food Res.* **2022**, *10*, 100437. [CrossRef]
19. Li, Z.; Wang, B.; Zhu, B.; Wang, Q.; Zhu, W. Thermal Error Modeling of Electrical Spindle Based on Optimized ELM with Marine Predator Algorithm. *Case Stud. Therm. Eng.* **2022**, *38*, 102326. [CrossRef]
20. Katılmış, Z.; Karakuzu, C. ELM Based Two-Handed Dynamic Turkish Sign Language (TSL) Word Recognition. *Expert Syst. Appl.* **2021**, *182*, 115213. [CrossRef]
21. Diker, A.; Avci, E.; Tanyildizi, E.; Gedikpinar, M. A Novel ECG Signal Classification Method Using DEA-ELM. *Med. Hypotheses* **2020**, *136*, 109515. [CrossRef]
22. He, C.; Wu, T.; Gu, R.; Jin, Z.; Ma, R.; Qu, H. Rolling Bearing Fault Diagnosis Based on Composite Multiscale Permutation Entropy and Reverse Cognitive Fruit Fly Optimization Algorithm—Extreme Learning Machine. *Measurement* **2021**, *173*, 108636. [CrossRef]
23. Huang, G.-B.; Zhou, H.; Ding, X.; Zhang, R. Extreme Learning Machine for Regression and Multiclass Classification. *IEEE Trans. Syst. Man Cybern. Part B-Cybern.* **2012**, *42*, 513–529. [CrossRef] [PubMed]
24. Hu, W.; Chang, H.; Gu, X. A Novel Fault Diagnosis Technique for Wind Turbine Gearbox. *Appl. Soft Comput.* **2019**, *82*, 105556. [CrossRef]
25. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.