Research Article

Data-Driven Fault Diagnosis for Rolling Bearing Based on DIT-FFT and XGBoost

Chuan Xiang, Zejun Ren, Pengfei Shi, and Hongge Zhao

College of Marine Electrical Engineering, Dalian Maritime University, Dalian 116026, China

Correspondence should be addressed to Chuan Xiang; cxiang@dlmu.edu.cn

Received 17 April 2021; Accepted 18 May 2021; Published 29 May 2021

1. Introduction

Rolling bearing is an extremely important basic mechanical component of rotating machinery. It is widely used in various fields of national economy and defense due to its high efficiency, easy assembly, and lubrication [1]. The health state of rolling bearing is directly related to the performance and service life of mechanical equipment. According to an incomplete statistic, about 30% of rotating machinery faults are caused by rolling bearing [2]. Therefore, it is necessary to diagnose and identify the fault of rolling bearing timely.

Over the years, some methods and techniques have been used to monitor the health state of equipment [3–7]. For the fault diagnose of bearing, most of them are based on the analysis of the vibration signal of bearing [8–13]. Generally, most of these methods consist of two stages: data processing and fault state determination [11]. For data processing, most methods extract the fault parameters of vibration signal from the amplitude spectrum, amplitude-frequency diagram, power spectrum, or wavelet spectrum in the time domain and frequency domain and then constitute eigenvector with these parameters for signal analysis. Signal analysis methods mainly include the time domain analysis, frequency domain analysis, and time-frequency domain analysis [12, 13]. Time domain analysis approaches analyze the vibration signal as a function of time [14], such as the spike energy method [15] and the signal enveloping method [16]. Frequency domain analysis, based on the availability of the Fourier transform technology, extracts the vibration signal features more easily than the time domain analysis [16]. The Fourier transform includes short-time Fourier Transform, Fast Fourier Transform (FFT), and Discrete Fourier Transform (DFT). Among them, FFT can perform DFT on finite sequence quickly. Thus it has an excellent performance in the feature extraction of fault diagnosis [14, 17, 18]. Time-frequency domain analysis approaches combine both the time and frequency domain information to study the inner features of signal, such as the Gabor transform [19], continuous wavelet transform [20], and the Wigner–Ville distribution [21]. There are also some other signal processing methods used to construct feature sets, such as sample entropy, fuzzy entropy, and amplitude spectral entropy [11].

For fault state determination, some different Machine Learning (ML) approaches are used to construct a classifier,
for instance, traditional ML, Deep Learning (DL), and Ensemble Learning (EL). In aspect of traditional ML, Chour et al. [8] applied SVM to automate the fault diagnosis procedure. It extracted the feature of the vibration signal of faulty bearing with Alpha-stable distribution. Bu et al. [22] proposed a method combining LS-SVM and Local Mean Decomposition (LMD) to diagnose the bearing fault. The original vibration signal was decomposed by LMD and trained to constitute the feature vectors. LS-SVM was used to determine the health state of bearing. However, the diagnosis accuracies of these traditional methods were only about 90%. In aspect of DL, Yang et al. [10] constructed a data set after extracting the features of vibration signal in the frequency domain. The data set was used to train a deep neural network (DNN) to classify the fault types. Jiang et al. [23] presented a method based on the convolutional neural network (CNN). The feature parameters of Mel-frequency cepstral coefficients and delta cepstrum were extracted to train the diagnosis model. However, parameter adjustment of these DL methods is difficult and the training time is usually very long. In aspect of EL, Hu et al. [24] proposed a method combining kernel principle component analysis with random forest (RF); a group of classifiers were trained in high dimensional kernel space with RF method. But the overfitting occurs for RF when the data has much noise. A new algorithm of EL called eXtreme Gradient Boosting (XGBoost) was proposed by Chen [25]. This method has many advantages, such as regularization, parallel processing, and missing value processing. It has excellent performance on regression and classification issues [26, 27]. Rao et al. [28] applied the XGBoost algorithm to detect anomalies of the steam turbine based on learning historical data and got a promising result.

This study is to propose an approach based on FFT with Decimation-In-Time (DIT) and XGBoost to diagnose and identify the rolling bearing faults. The experimental data is provided by the Case Western Reserve University Bearing Data Center (CWRU). The DIT-FFT is used to process vibration signal, and the XGBoost model is used as a classifier to diagnose the faults of rolling bearing.

2. Related Algorithms

2.1. DIT-FFT. FFT is a fast computation method of DFT. This section starts with an introduction of DFT. Assuming \( x(n) \) is a typical \( N \)-points finite sequence, then its DFT is defined as

\[
X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk},
\]

where \( k \) is the normalized digital frequency, \( k = 0, 1, \ldots, N - 1 \), \( W_N^{nk} = e^{-j2\pi nk/N} \). Unfolding equation (1) into the matrix form, we can obtain

\[
\begin{bmatrix}
X(0) \\
X(1) \\
\vdots \\
X(N-1)
\end{bmatrix} =
\begin{bmatrix}
W_N^0 & W_N^1 & \cdots & W_N^{N-1}
\end{bmatrix}
\begin{bmatrix}
x(0) \\
x(1) \\
\vdots \\
x(N-1)
\end{bmatrix}.
\]

(2)

Its abbreviated matrix form is

\[
X = Fx,
\]

where \( X \) is the DFT matrix, \( x \) is the sequence matrix, and \( F \) denotes the transformation matrix. It can be seen from \( F \) that the computational complexity is \( N \times N \) recorded as \( O(N^2) \) for a \( N \)-point DFT. It includes \( N \times N \) times multiplication and \( N \times (N - 1) \) times addition. Thus, it will consume a lot of computing time and memory space to process data directly by this method. This study adopted the radix-2 FFT algorithm [29] to improve computational efficiency and reduce computational complexity. In practical applications, there are many different radix-2 FFT algorithms. Among them, the radix-2 FFT algorithm with DIT (radix-2 DIT-FFT) is the most important one. Its flow diagram of the butterfly operation theorem is shown in Figure 1.

As can be seen from Figure 1, the output of the upper branch equals the sum of the inputs of the upper branch and the lower branch, that is, \( X(k) = X_0(k) + W_N^{k}X_1(k) \). The output of the lower branch equals the difference between the inputs of the upper branch and the lower branch, that is, \( X(k + (N/2)) = X_0(k) - W_N^{k}X_1(k) \). With the upper form, for \( N = 2^l \) (\( l \) is a positive integer), the lengths of the subsequence after the first, the second, \ldots, and the last decompositions are \( 2^{l-1}, 2^{l-2}, \ldots, \) and \( 2^0 = 1 \), respectively. Therefore, the total number of decompositions is \( l = \log_2 N \). For each decomposition, the times of multiplication and addition of the corresponding butterfly operations are \( N/2 \) and \( N \), respectively. Thus, the total multiplication times \( M_c \) (i.e., computational complexity) and addition times \( A_c \) are as follows:
\[ X_0 (k) \longleftrightarrow X (k) = X_0 (k) + W_N^k X_1 (k) \]
\[ X_1 (k) \longleftrightarrow W_N^k \]
\[ 0 \leq k \leq N/2 - 1 \]
\[ X (k + N/2) = X_0 (k) - W_N^k X_1 (k) \]

**Figure 1:** Flow diagram of butterfly operational theorem.

\[ M_c = \frac{N}{2} I = \frac{N}{2} \log_2 N, \]
\[ A_c = NI = N \log_2 N. \]  

(4)

(5)

The ratio of multiplications of DIT-FFT and DFT \( \alpha_M \) (i.e., the ratio of computational complexity) and the ratio of additions \( \alpha_A \) are

\[ \alpha_M = \frac{M_c}{N^2} = \frac{(N/2) \log_2 N}{N^2} = \frac{1}{2N} \log_2 N, \]
\[ \alpha_A = \frac{A_c}{N^2} = \frac{N \log_2 N}{N^2} = \frac{1}{N} \log_2 N. \]  

For the radix-2, DIT-FFT and DFT algorithm with different \( N \), \( \alpha_M \) and \( \alpha_A \) were illustrated in Figure 2. As we can see from Figure 2, both of \( \alpha_M \) and \( \alpha_A \) decrease rapidly with the increase of \( N \). In other words, the DIT-FFT method accelerates the calculation speed and improves the calculation efficiency obviously when \( N \) is large enough. But when \( N \) is greater than \( 2^{11} \), \( \alpha_M \) and \( \alpha_A \) decrease slowly. Therefore, this paper chose \( N = 2^{11} \) as the length of each group sample to reduce the computational complexity.

2.2. XGBoost. XGBoost is an algorithm of EL, which is based on Classification and Regression Tree (CART) [25]. Its objective function is defined as

\[ \text{Obj}^{(i)} = \sum_{i=1}^{n} L(y_i, \hat{y}^{(i-1)}_i) + f_i(x_i) + \Omega (f_i) + \text{constant}, \]  

(6)

where \( L \) is a differentiable convex loss function that measures the difference between the prediction \( \hat{y}_i \) and the target \( y_i \). \( f_i(x_i) \) is the final score function of the sample in the \( i \)th round and \( x_i \) is the \( i \)th input of the sample. It can be expressed as

\[ f_i(x) = \omega_{q(x)} q: R^m \rightarrow [1, 2, \ldots, T], \omega \in R^T, \]  

(7)

where \( \omega \) is the leaf vector, \( q \) is the structure of the tree, and \( T \) represents the number of leaf nodes.

The second term \( \Omega \) on the right side of equation (6) penalizes the complexity of the model. It is defined as

\[ \Omega (f) = \gamma T + \frac{1}{2} \lambda \| \omega \|^2, \]  

(8)

where \( \lambda \) and \( \gamma \) are two model parameters used to control the proportion of \( \Omega \). \( \| \omega \|^2 \) represents \( L_2 \) regularization processing on \( \omega \), it can also be expressed as \( \sum \omega^2 \). If the value of \( \Omega (f) \) is small, the complexity of the tree is low while the generalization ability is strong.

After the second-order Taylor expansion of equation (6), a new objective function is obtained:

\[ \text{Obj}^{(i)} = \sum_{i=1}^{n} \left[ L(y_i, \hat{y}^{(i-1)}_i) + g_i f_i(x_i) + \frac{1}{2} h_i f_i^2 (x_i) \right] + \Omega (f_i) + \text{constant}, \]  

(9)

where \( g_i = \partial_{\hat{y}^{(i-1)}_i} L(y_i, \hat{y}^{(i-1)}_i) \), \( h_i = \partial_{\hat{y}^{(i-1)}_i}^2 L(y_i, \hat{y}^{(i-1)}_i) \) are the first- and second-order gradient statistics on the loss function \( L \), respectively.

The final objective function only depends on the first-order and the second-order derivative of the error of each data point. Removing the constant term and expanding \( \Omega (f_i) \), the objective function is updated to

\[ \text{Obj}^{(i)} = \sum_{i=1}^{n} \left[ g_i \omega_{q(x)} + \frac{1}{2} h_i \omega_{q(x)}^2 \right] + \gamma T + \lambda \sum_{j=1}^{T} \omega_j^2, \]  

(10)

where \( I_j = \{ i | q(x_i) = j \} \) is defined as the instance set of leaf \( j \).

For the sake of simplicity, we define \( G_j = \sum_{i \in I_j} g_i \) and \( H_j = \sum_{i \in I_j} h_i \); then,

\[ \text{Obj}^{(i)} = \sum_{j=1}^{T} \left[ G_j \omega_j + \frac{1}{2} (H_j + \lambda) \omega_j^2 \right] + \gamma T. \]  

(11)

Next computing partial derivatives of \( \omega_j \) and bringing the results back to equation (11), the following formula can be obtained:

\[ \text{Obj} = \frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T. \]  

(12)
It can be used as a scoring function to measure the quality of a tree structure $q$. The smaller Obj is, the well-structured the tree will be. This score is like the impurity score for evaluating decision trees, except that it is derived for a wider range of objective functions. Generally, it is impossible to enumerate all possible tree structures $q$. A greedy algorithm that starts from a single leaf and iteratively adds branches to the tree is used instead [25]. Let $I_L$ and $I_R$ be the instance sets of left and right nodes after the split, respectively. Define $I = I_L \cup I_R$, the loss reduction after the split is given by

$$L_{\text{split}} = \frac{1}{2} \left[ \frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda} \right] - \gamma.$$  

(13)

This formula is usually used in practice for the evaluation of the split candidate node.

### 3. Bearing Fault Test Bench

In this paper, the proposed DIT-FFT-XGBoost method is tested by the SKF rolling bearing data sets of CWRU which are standard data for testing diagnosis methods of bearing fault. Figure 3 illustrates the bearing fault test bench of CWRU. It consists of a 2hp electric motor, a torque transducer/encoder, a dynamometer, and the control electronics. Generally speaking, the rolling element (RE), inner race (IR), and outer race (OR) of the rolling bearing of the electric motor are easily damaged. The test bench can simulate these fault types, that is, RE fault, IR fault, and OR fault. For the OR fault, the damage positions are located at 3 o'clock, 6 o'clock, or 12 o'clock. In this test bench, there are three acceleration sensors used to record the vibration accelerations signals of the rolling bearing. They are placed on the rack with a magnetic base (BA), the bearing seat of the fan end (FE), and the drive end (DE), respectively. The vibration acceleration signals are collected by a 16-channel data recorder. The sampling frequency of FE is 12 kHz, and the fault sampling frequency of DE is 12 kHz or 48 kHz. The bearing speed is 1730, 1750, 1772, or 1797 rpm.

The test bench of CWRU has recorded huge amounts of data. This paper only chose the data of 12 k DE and 1772 rpm for fault diagnosis. There are 9 types of bearing fault and one normal state. The 9 fault types are the RE fault, IR fault, and OR fault with 0.007-, 0.014-, and 0.021-inch damage diameter, respectively. The data of OR fault is only taken from the damage position at 6 o’clock. The data used in this paper is listed in Table 1.

### 4. Bearing Fault Diagnosis

#### 4.1. Fault Diagnosis Procedure.

The fault diagnosis procedure of rolling bearing with the DIT-FFT-XGBoost method is shown in Figure 4. It mainly includes two stages: data processing and fault state determination. There are two steps in the data processing stage. Step 1: calculate the standard deviation and average value of input data and then standardize the data to be normal distribution to eliminate the impact of the abnormal samples on the data processing. It is suitable for complicated big data. Step 2: process the standardized data with DIT-FFT, and then divide them into the training set and test set. In the fault state determination stage, feed the training set into the XGBoost model. The XGBoost model trains the negative gradient of loss function each time to adjust its parameters. Finally, the well-trained model identifies the fault type of bearing. Analysis was performed with the Python system for computing on a laptop with Intel (R) Core (TM) i5-7200 CPU @ 2.70 GHz.

#### 4.2. Data Processing.

The original data of RE fault, IR fault, and OR fault with the 0.007-inch damage diameter collected by the recorder at DE under the conditions of 12 kHz and 1772 rpm are shown in Figure 5.

After fitting and standardizing the data of the above three fault types to be a normal distribution, the standardization results are shown in Figure 6.

Next, transform the standardization results by DIT-FFT; the absolute values of them are shown in Figure 7.

The processing results of the normal state and 9 fault types with 0.014- and 0.021-inch damage diameters are illustrated in Figures 8–14, respectively.

For each of the above figures, the waveform from top to bottom are original data, standardization result, and transformation result, respectively. As mentioned above, $N = 2^{11} = 2048$ data length was chosen as a group sample for the radix-2 DIT-FFT algorithm in this paper. Data length is 2048 for all of the above data processing figures. According to the data lengths of the 12 k DE bearing data as listed in Table 1 and the above data length selection rule of sample, we

---

**Table 1:** Bearing data at DE under the conditions of 12 kHz and 1772 rpm

| Damage diameter (inches) | Fault type | Data length |
|-------------------------|------------|-------------|
| 0.007                   | RE         | 121410      |
|                         | IR         | 121991      |
|                         | OR         | 122426      |
| 0.014                   | RE         | 122136      |
|                         | IR         | 121846      |
|                         | OR         | 122136      |
| 0.021                   | RE         | 121410      |
|                         | IR         | 121991      |
|                         | OR         | 122426      |
| 0                       | Normal state | 483903     |
**Figure 4:** Diagnosis procedure of DIT-FFT-XGBoost.

**Figure 5:** The original data of (a) RE fault, (b) IR fault, and (c) OR fault with 0.007-inch damage diameter.

**Figure 6:** The standardization results of RE fault (a), IR fault (b), and OR (c) with 0.007-inch damage diameter.
constructed the training set and test set as illustrated in Table 2. For each bearing fault type and the normal state, 50 group samples were taken from original data as the training set, that is, data length of 102,400. The remaining data was used to construct the test set. Thus, there were 9 group samples as test set for each fault type (i.e., data length of 18,432), and 186 group samples as the test set for the normal state (i.e., data length of 380,928). Therefore, the training set and test set were composed of 500 and 267 group samples, respectively.

4.3. Fault State Determination. In order to make full use of the training set to improve the diagnosis accuracy of the
Figure 10: 0.014-inch IR fault.

Figure 11: 0.014-inch OR fault.

Figure 12: 0.021-inch RE fault.

Figure 13: 0.021-inch IR fault.
XGBoost model, the \( k \)-fold cross-validation was introduced to divide the training set into \( k \) folds. Among them, \( k - 1 \) folds were used for the training model each time, and the remaining one was used for the testing model. When the parameters of the XGBoost model were default values, the influence of \( k \) on the training time and diagnosis accuracy of the model is illustrated in Figure 15. With the increase of \( k \) from 3 to 7, the training time gradually increases from 31.91 s to 89.62 s, but the diagnosis accuracy increases firstly and then decreases. When \( k = 5 \), the diagnosis accuracy is the highest; meanwhile, the training time is not too long. Therefore, the training set is divided into 5 folds.

There are several parameters of the XGBoost model that affect the accuracy of fault diagnosis. Their detailed information is listed in Table 3.

Among them, the most important parameters are the \text{max\_depth}, \text{min\_child\_weight}, and \text{gamma}. The \text{max\_depth}, that is, the maximum depth of a tree, is a parameter that needs to be adjusted when building a tree. If it is too large, the model will be more complicated and easier to overfit [30]. The test value of this parameter was selected from 3 to 6 in this paper. \text{Min\_child\_weight}, that is, minimum sum of instance weight needed in a child (\( H_j \) in (15)), is used to avoid overfitting. If the tree partition step results in a leaf node with the sum of instance weight less than \text{min\_child\_weight}, the building process will give up further partition [30]. The larger the \text{min\_child\_weight} is, the more conservative the model will be. Its range was from 1 to 5. The influence of these two parameters on the diagnosis accuracy of the XGBoost model is shown in Figure 16.

As can be seen from Figure 16, on the whole, the diagnosis accuracies of the XGBoost model for different \text{max\_depths} decrease with the increase of \text{min\_child\_weight} from 1 to 5. The diagnosis accuracy is the highest when \text{max\_depth} = 4. In particular, the model has the highest diagnosis accuracy (up to 100\%) when \text{min\_child\_weight} = 1.

The last important parameter \text{gamma} (\( \gamma \)) represents the minimum loss reduction required to make a further partition on a leaf node of the tree (\( \gamma \) in equation (8)). The larger \( \gamma \) is, the more conservative the algorithm will be [30]. So, its range was from 0 to 0.5 in this study. The influence of \( \gamma \) on the diagnosis accuracy of the XGBoost model is shown in Figure 17.

As can be seen from Figure 17, the diagnosis accuracy of the XGBoost model decreases slightly with the increase of \( \gamma \) from 0 to 0.5. When \( \gamma = 0 \) or 0.1, the model has the highest accuracy, up to 100\%. Moreover, compared with \( \gamma = 0.1 \), the calculation of the model is small when \( \gamma = 0 \); thus, \( \gamma = 0 \) is the better choice for the XGBoost model.

4.4. Result and Analysis. T-distributed stochastic neighbor embedding (T-SNE) is a ML algorithm for dimensionality reduction. It is convenient for the visualization of data. This paper adopted this algorithm to reduce each group sample to
2D and visualize the effect of data processing with DIT-FFT. Figure 18 shows the outcome of dimensionality reduction of processed data. It can be seen that the effect of data processing with DIT-FFT is very obvious. Each type of fault clusters together after the data processing.

When the parameter values of the XGBoost model are default in Table 3, the original data and processed data with DIT-FFT are used to train the XGBoost model and diagnose the test set, respectively. The comparisons of training time of model and diagnosis accuracy of test set between the above two cases are listed in Table 4.

It can be seen from Table 4 that, using the original data as the input of the XGBoost model, the training time is 52.65 s. But it is drastically reduced to 31.91 s using the data processed by DIT-FFT. The diagnosis accuracy of the test set has also greatly improved from about 55% to 93%.

Figure 15: The influence of $k$ on training time and diagnosis accuracy of the model.

Table 3: Main parameters information of XGBoost

| Number | Parameter      | Implication                                           | Default value |
|--------|----------------|-------------------------------------------------------|---------------|
| 1      | max_depth      | Maximum depth of a tree                               | 6             |
| 2      | gamma ($\gamma$) | Minimum loss function decline value                   | 0             |
| 3      | max_delta_step | Maximum delta step we allow each leaf output to be    | 0             |
| 4      | lambda ($\lambda$) | $L_2$ regularization term on weights                  | 1             |
| 5      | alpha ($\alpha$) | $L_1$ regularization term on weights                  | 0             |
| 6      | min_child_weight | Minimum sum of instance weight needed in a child     | 1             |
| 7      | Eta            | Step size shrinkage used in update to prevent overfitting | 0.3           |
| 8      | Subsample      | Subsample ratio of the training instances             | 1             |

Figure 16: The influence of max_depth and min_child_weight on diagnosis accuracy.
Next, the diagnosis accuracy of the XGBoost model without parameter adjustment for each fault type is further analyzed. The confusion matrices associated with the obtained results above are illustrated in Figure 19. As can be seen from Figure 19(a), when the XGBoost model is trained with the original data, it only has high accuracy for the OR fault with a 0.021-inch damage diameter (i.e., label 8). But when the model is trained with the data processed by DIT-FFT, it has high accuracies for 7 fault types (i.e., labels 0, 1, 2, 3, 5, 8, and 10), as shown in Figure 19(b). Thus, it further shows that data processed by DIT-FFT can greatly improve the diagnosis accuracy of bearing.

To further improve the accuracy of model diagnosis, the parameters of the XGBoost model are adjusted to the best values, that is, max_depth = 4, min_child_weight = 1, and γ = 0. Both the train set and test set are processed by DIT-FFT previously. The confusion matrix for the test set is shown in Figure 20.

We can see from Figure 20 that the XGBoost model with the best parameter values has high accuracies for 9 fault types (i.e., labels 0, 1, 2, 3, 4, 5, 7, 8, 9, and 10). The total diagnosis accuracy of the test set is up to 98.12%. But the model has a low diagnosis accuracy for the RE fault with 0.021-inch damage diameter (i.e., label 6). It misidentifies this fault as the RE fault with a 0.014-inch damage diameter (i.e., label 3). Analyzing the vibration signals of these two faults, we find that they are very similar so that the model cannot identify them accurately. This problem will be further solved in future work.
5. Method Comparison

In this part, the proposed method DIT-FFT-XGBoost is compared with other methods from two aspects, for example, data processing and fault state determination. In aspect of data processing, the Empirical Mode Decomposition (EMD) method was chosen to compare with the DIT-FFT method. EMD has a very good performance in dealing with nonstationary and nonlinear data [17, 27]. It decomposes signal based on the time scale characteristics of the data itself. Taking the RE fault with 0.007-inch damage diameter (i.e., label 0) as an example, the decomposition result of vibration signal by EMD method is illustrated in Figure 21.

After the analysis of the kurtosis of Intrinsic Mode Function (IMF) components shown in Figure 21, four components IMF1 ∼ IMF4 are chosen to build a new dataset as the input data to train the XGBoost model. The diagnosis accuracy of this well-trained model for the label-0 fault is only 82.40%. As a comparison, when using the vibration signal of the label-0 fault processed by DIT-FFT to train the XGBoost model, the diagnosis accuracy increases up to 88.9%, which is higher than that of EMD.

![Figure 19: Confusion matrixes obtained from (a) original data and (b) processed data.](image1)

![Figure 20: Confusion matrix for the test set.](image2)
In aspect of fault state determination, the SVM method in ML; GBDT and RF method in EL; and DNN and CNN in DL are chosen to compare with XGBoost. All of the input training data of these models are preprocessed by DIT-FFT. The comparisons of these models in terms of training time and diagnosis accuracy are shown in Table 5.

As we can see from Table 5, the training time of SVM is the shortest, but its diagnosis accuracy is lower than 90%. It does not satisfy the requirements of the high diagnosis accuracy of bearing faults. Compared with other methods, XGBoost has the highest diagnosis accuracy (up to 98.12%) without a significant increase in training time (59.53 s). Therefore, it is the best choice for the diagnosis of bearing faults.

6. Conclusion

This paper proposed an approach based on DIT-FFT and XGBoost for the fault diagnosis of rolling bearing. The DIT-FFT was used to process vibration signal, and the XGBoost model was used as a classifier for fault identification. The proposed approach was tested with CWRU bearing data and compared with some common methods. The following conclusions can be drawn:

(1) After the vibration signal was processed by DIT-FFT, both of the training time and diagnosis accuracy of the XGBoost model had a significant improvement. The training time reduced from 52.65 s to 31.91 s; meanwhile, the diagnosis accuracy of the test set improved from about 55% to 93%.

(2) After the parameters of the XGBoost model were adjusted to the best values, the model had high accuracies for 9 fault types and the total diagnosis accuracy of the test set was further up to 98.12%.

(3) Compared with the data processing method of EMD, DIT-FFT can extract more fault features and had higher diagnosis accuracy.

(4) Compared with other traditional ML, EL, and DL methods, XGBoost had the highest diagnosis accuracy (up to 98.12%) without a significant increase in training time (59.53 s).

There is no doubt that the combination of DIT-FFT and XGBoost can be used to diagnose bearing faults quickly and accurately.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Table 5: Comparison of different models for fault diagnosis

| Item          | Machine Learning | EL | DL | XGBoost |
|---------------|------------------|----|----|---------|
| Training time (s) | 10.16            | 48.21 | 52.17 | 132.56 | 50.11 | 59.53 |
| Accuracy (%)    | 89.88            | 11.23 | 93.25 | 83.52 | 95.13 | 98.12 |

Figure 21: Data processing with EMD.
Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 51507025, Natural Science Foundation of Liaoning Province under Grant 20180550822, and Fundamental Research Funds for the Central Universities under Grant 3132019014.

References

[1] R. B. Randall, Vibration-based Condition Monitoring: Industrial, Aerospace and Automotive Applications, John Wiley & Sons, Hoboken, NJ, USA, 2011.

[2] F. Immovilli, C. Bianchini, M. Cocconcelli, A. Bellini, and R. Rubini, “Bearing fault model for induction motor with externally induced vibration,” IEEE Transactions on Industrial Electronics, vol. 60, no. 8, pp. 3408–3418, 2013.

[3] C. Wei, Z. Shen, D. Xiao, L. Wang, X. Bai, and H. Chen, “An optimal scheduling strategy for peer-to-peer trading in interconnected microgrids based on RO and Nash bargaining,” Applied Energy, vol. 275, 2021.

[4] C. Wei, M. Benosman, and T. Kim, “Online parameter identification for state of power prediction of lithium-ion batteries in electric vehicles using extremum seeking,” International Journal of Control, Automation and Systems, vol. 17, no. 11, pp. 2906–2916, 2019.

[5] L. Wang, R. Yan, F. Bai, T. Saha, and K. Wang, “A distributed inter-phase coordination algorithm for voltage control with unbalanced PV integration in LV systems,” IEEE Transactions on Sustainable Energy, vol. 11, no. 4, pp. 2687–2697, 2020.

[6] F. Xiao, Q. Li, and K. Wang, “Waste plastic triboelectric nanogenerators using recycled plastic bags for power generation,” ACS Applied Materials & Interfaces, vol. 13, no. 1, pp. 400–410, 2021.

[7] F. Xiao, Y. Zhang, L. Kang et al., “Integrated energy storage system based on triboelectric nanogenerator in electronic devices,” Frontiers of Chemical Science and Engineering, vol. 15, no. 2, pp. 238–250, 2021.

[8] B. Chouriri, M. Fabrice, A. Dandache, M. E. L. Arousii, and R. Saadane, “Bearing fault diagnosis based on Alpha-stable distribution feature extraction and SVM classifier,” in Proceedings of the International Conference on Multimedia Computing and Systems (ICMCS), pp. 1545–1550, Marrakesh, Morocco, April 2014.

[9] W. Mao, W. Feng, and X. Liang, “A novel deep output kernel learning method for bearing fault structural diagnosis,” Mechanical Systems and Signal Processing, vol. 117, pp. 293–318, 2019.

[10] Y. Yang, P. Fu, and Y. He, “Bearing fault automatic classification based on deep learning,” IEEE Access, vol. 6, pp. 71540–71554, 2018.

[11] Y. Tian, Z. Wang, and C. Lu, “Self-adaptive bearing fault diagnosis based on permutation entropy and manifold-based dynamic time warping,” Mechanical Systems and Signal Processing, vol. 114, pp. 658–673, 2019.

[12] F. Q. Li, Y. G. Fan, and Y. X. Zhang, “Rolling bearing fault detection method based on wavelet packet energy spectrum-PLS,” Advanced Materials Research, vol. 971–973, pp. 697–700, 2014.

[13] N. Wang and X. Liu, “Bearing fault diagnosis method based on hilbert envelope demodulation analysis,” IOP Conference Series: Materials Science and Engineering, vol. 436, no. 1, 2018.

[14] A. Issam, B. Nadir, F. Nadir, B. Oudjani, and A. Deliou, “Vibration-based bearing fault diagnosis by an integrated DWT-FFT approach and an adaptive neuro-fuzzy inference system,” in Proceedings of the 2013 3rd International Conference on Control, Engineering & Information Technology (CEIT), pp. 1–6, Tlemcen, Algeria, May 2013.

[15] M. Xu, “Spike EnergyTM and its applications,” The Shock and Vibration Digest, vol. 27, no. 3, pp. 11–17, 1995.

[16] D. Laha and P. Mandal, Handbook of Computational Intelligence in Manufacturing and Production Management, Information Science Reference, New York, NY, USA, 2007.

[17] V. K. Rai and A. R. Mohanty, “Bearing fault diagnosis using FFT of intrinsic mode functions in Hilbert-Huang transform,” Mechanical Systems and Signal Processing, vol. 21, no. 6, pp. 2607–2615, 2007.

[18] J. Xie, G. Du, C. Shen, N. Chen, L. Chen, and Z. Zhu, “An end-to-end model based on improved adaptive deep belief network and its application to bearing fault diagnosis,” IEEE Access, vol. 6, pp. 63584–63596, 2018.

[19] J. Luo, G. Liu, Z. Huang, and S. S. Law, “Mode shape identification based on Gabor transform and singular value decomposition under uncorrelated colored noise excitation,” Mechanical Systems and Signal Processing, vol. 128, pp. 446–462, 2019.

[20] A. Maamar, G. P. Bevan, P. A. Wallace, D. K. Harrison, and K. P. Ramachandran, “Fault diagnosis of a centrifugal pump using MLP-GABP and SVM with CWT,” Engineering Science and Technology, vol. 22, no. 3, pp. 854–861, 2019.

[21] Y. Wu, X. Li, and Y. Wang, “Extraction and classification of acoustic scattering from underwater target based on Wigner-Ville distribution,” Applied Acoustics, vol. 138, pp. 52–59, 2018.

[22] Y. Bu, J. Wu, J. Ma, X. Wang, and Y. Fan, “The rolling bearing fault diagnosis based on LMD and LS-SVM,” in Proceedings of the 26th Chinese Control and Decision Conference (2014 CCDCC), pp. 3797–3801, IEEE, Changsha, China, May-June 2014.

[23] Q. Jiang, F. Chang, and B. Sheng, “Bearing fault classification based on convolutional neural network in noise environment,” IEEE Access, vol. 7, pp. 69795–69807, 2019.

[24] Q. Hu, C. Sun, and L. Du, “Transformer fault diagnosis method using random forests and kernel principle component analysis,” High Voltage Engineering, vol. 36, no. 7, pp. 1725–1729, 2010.

[25] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp. 785–794, San Francisco, CA, USA, August 2016.

[26] J. Nobre and R. F. Neves, “Combining principal component analysis, Discrete wavelet transform and XGBoost to trade in the financial markets,” Expert Systems With Applications, vol. 125, pp. 181–194, 2019.

[27] Y. Zhang and J. Zhu, “Atrial fibrillation detection based on EEMD and XGBoost,” Journal of Physics: Conference Series, vol. 1229, no. 1, 2019.

[28] K. Rao, D. N. Kim, and J. J. Hwang, Fast Fourier Transform Algorithms and Applications, China Machine Press, Beijing, China, 2016.

[29] C. J. W. Tukey, “An algorithm for the machine calculation of complex Fourier series,” Mathematics of Computation, vol. 19, pp. 297–301, 1965.

[30] “XGBoost parameters,” https://xgboost.readthedocs.io/en/latest/parameter.html.