

Research Article

An Accurate Prediction Algorithm of RUL for Bearings: Time-Frequency Analysis Based on MRCNN

Dezhi Huang ¹, Gang Yu,² Jian Zhang,¹ Jian Tang,³ and Jian Su ¹

¹School of Computer and Software, Nanjing University of Information Science & Technology, 210044, China

²BGRIMM Technology Group Co., Ltd., Beijing 102600, China

³Faculty of Information Technology, Beijing University of Technology, Beijing 100024, China

Correspondence should be addressed to Jian Su; sj890718@gmail.com

Received 3 April 2022; Accepted 31 May 2022; Published 20 June 2022

Academic Editor: Kalidoss Rajakani

Copyright © 2022 Dezhi Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Remaining useful life (RUL) prediction of mechanical components is of high research value in the field of prognostics and health management (PHM). However, RUL prediction problems are completely challenging due to the complicity of bearings' operating environment. In this paper, we transform the vibration acceleration signal collected by sensors into a time-frequency domain matrix through continuous wavelet transform (CWT) and then extract the features of the time-frequency domain matrix through the proposed multiscale residual convolutional neural network (MRCNN), which enables the model to extract more local and global features while constructing more accurate health indicators (HI). In order to highlight the degradation trend of mechanical components, the obtained health indicators are smoothed by exponential moving average (EMA). Finally, linear regression is exploited to predict the RUL of the bearing. Performance evaluations based on the public dataset PRONOSTIA demonstrate the effectiveness of our proposed algorithm, which is superior to existing data-driven algorithms in terms of prediction accuracy.

1. Introduction

Bearing plays an irreplaceable role in the machinery industry, which always keeps running as the core component of the machinery, especially as a major equipment or a key basic component. However, maintaining their availability and reliability usually consumes significant resources, leading to increased enterprise costs. Consequently, the bearings' PHM becomes increasingly important and attractive in the industrial field, where accurate prediction of RUL of bearings can improve production efficiency and maximize economic benefits. Therefore, RUL prediction is essential for determining bearing conditions and developing maintenance strategies [1–4]. Fortunately, due to the development of the Internet of Things, massive amounts of bearing operating data on PHM systems could be collected for further data processing to estimate bearing states. However, how to effectively extract features from the data and accurately predict the remaining useful life is still a considered challenge for PHM [5–9], due to the harsh industrial environment and complex fault causes.

RUL prediction methods in bearings are mainly divided into two categories: model-based methods (also known as physics-based methods) and data-driven methods. Model-based methods accurately describe the degradation mechanism of the equipment by building mathematical-statistical models with a full understanding of the failure mechanism. Concurrently, the parameters of the model-based method can be estimated by the collected data [10]. However, due to the complex internal structure of the device and the distinct degradation mechanisms, it is impractical to rely solely on a large amount of a priori and expert knowledge to accurately predict the RUL [11–13]. Compared with model-based methods, data-driven methods do not require establishing complex mathematical models, which predict the degradation trend and the RUL of the equipment through using the data analysis method and finding out the intrinsic relationship of the degradation trend according to the historical data collected by the sensor. Furthermore, with the flourishing of the Internet of Things and the advent of the era of big data, massive amounts of useful data can be

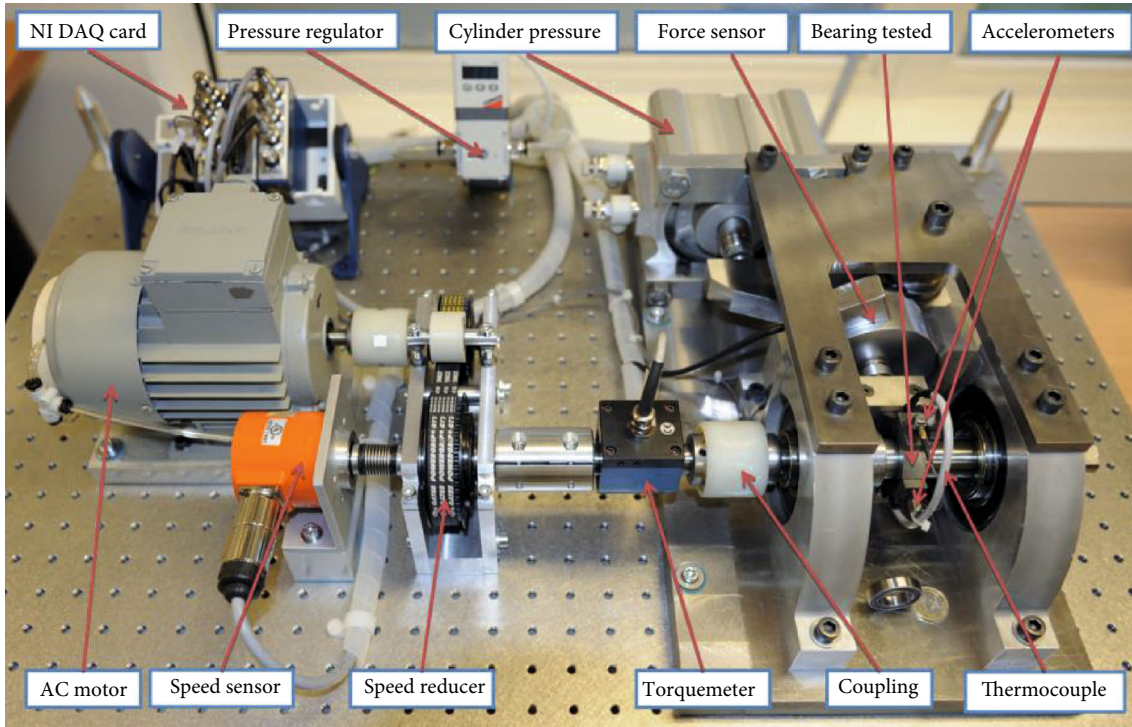


FIGURE 1: Bearing degradation test platform.

collected through monitoring sensors. In addition, owing to the advantages of low cost and high accuracy, data-driven methods are progressively becoming a current research hot-spot. Generally, the data-driven prediction method consists of three steps: data collection, HI construction, and RUL prediction [14]. Because of its powerful feature extraction and prediction capabilities, deep learning is widely used in PHM HI construction to predict the RUL without physical domain knowledge. Gebrael et al. [15] designed an artificial neural network data-driven model to predict the RUL of bearings, which is the initial successful attempt of deep learning in bearing RUL prediction. Zhao et al. [16] combined a convolutional neural network (CNN) with a bidirectional long short-term memory (Bi-LSTM) network to form a (CBLSTM) network to predict tool wear, where they utilized CNN to extract degraded features from the original signals as inputs for a Bi-LSTM network to construct HI. This improved deep learning model had good prediction accuracy; however, it also consumed more time. In [17], the Fourier transform for the original signal is exploited to obtain frequency information, and then, the deep neural network (DNN) based on stacked autoencoder (SAE) is employed for fault diagnosis of hydraulic pumps. Malhi et al. [18] adopted recurrent neural networks (RNN) to possess better prediction accuracy for long-term health condition prediction of machines. Peng et al. [19] proposed a Bayesian deep learning method to cope well with the RUL prediction uncertainty problem. From the relatively novel articles in recent years, data-driven methods have more significant advantages than model-based methods in terms of prediction accuracy, which is also a direction worthy of key research in the future.

Due to massive amounts of noise, extracting effective degradation features is always a pivotal challenge for RUL prediction. Actually, the bearings have to keep operating in a harsh industrial environment and frequently aggravate complex faults. In this paper, the experimental data are employed from PRONOSTIA in IEEE PHM 2012 Data Challenge [20]. The bearing degradation experimental platform is shown in Figure 1.

The platform uses sensors to collect vibration acceleration signal information during bearing degradation, where two types of vibration acceleration signals are collected: horizontal vibration acceleration signals and vertical vibration acceleration signals. The raw vibration signals for the full life of Bearing1_1 and Bearing1_2 are shown in Figure 2. The x -axis direction indicates the time in 10s, and the y -axis direction indicates the amplitude. The two different types of degradation can be clearly seen in Figure 2. One of the Bearing1_1 is the slow degradation type, and its amplitude shows a gradual increase with time. Seen from Figure 2, the amplitude increases slowly and regularly from times 0s to 27000s. From 27000s to 28030s, the amplitude increases faster and the bearing is damaged finally when the bearing reaches the end of its life. Bearing1_2 illustrates the type of sudden degradation with massive amounts of noise influence, whose trend of increasing amplitude from time 0s to 8300s is not obvious. From 8300s to 8710s, the amplitude suddenly increases sharply, which implies that the bearing is damaged finally. Consequently, Figure 2 shows that the slow degradation type is favourable for predicting RUL with expectation based on the growth of amplitude, while it is challenging with sudden degradation type of amplitude.

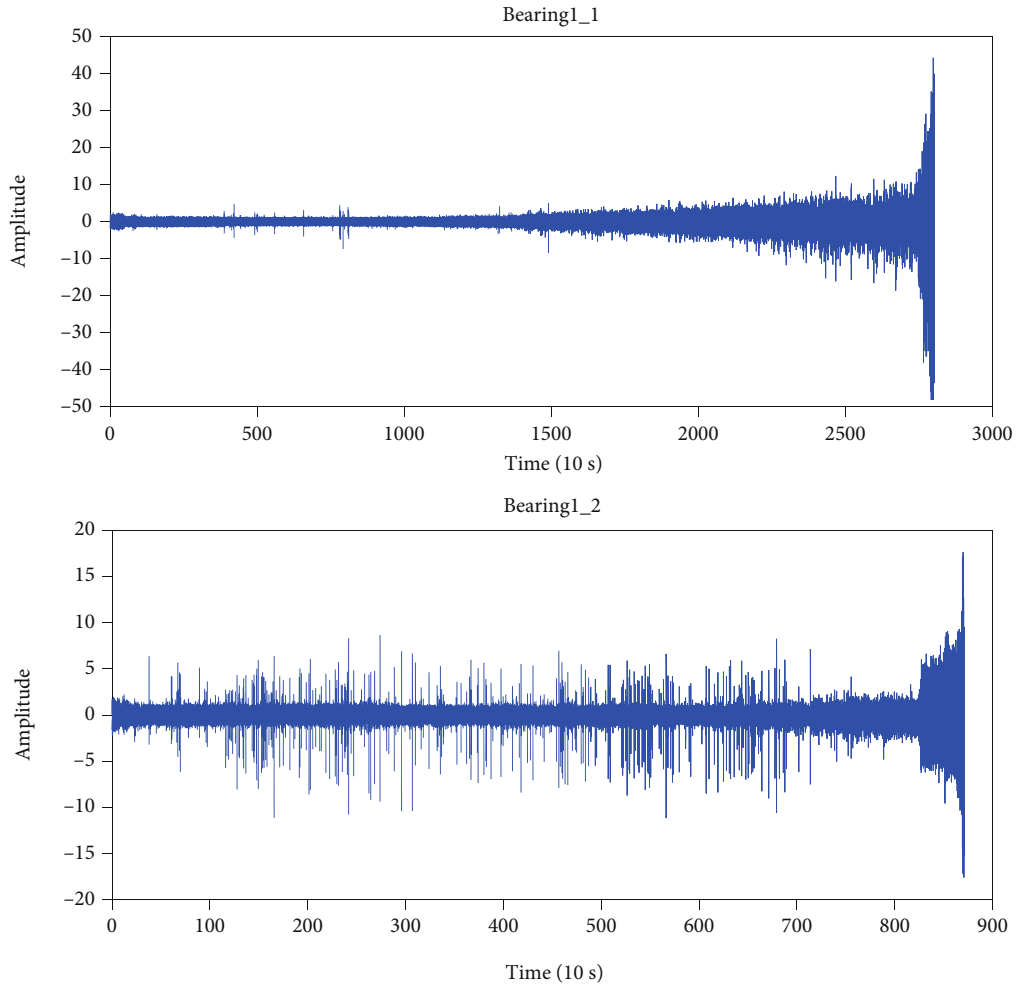


FIGURE 2: Raw vibration signal of Bearing1_1 and Bearing1_2.

The slow degradation bearings have the characteristic of stationary signal type, and the bearing's time-domain features can describe the corresponding variations well. However, many bearings are the nonstationary signal type, which forces researchers to find weak fault information in the background of strong noise. Fortunately, time-frequency analysis is an effective method for analyzing nonstationary and transient vibration signals. Consequently, a new deep learning method for RUL prediction based on continuous wavelet transform (CWT) and multiscale residual convolutional neural network (MRCNN) is proposed in this paper. Firstly, the vibration acceleration signals collected from the bearings are processed by continuous wavelet transform and converted into a time-frequency domain 2-D matrix. In order to cut down the input volume of the model, the size of the 2-D matrix is reduced by exploiting the nearest interpolation algorithm. Afterward, the resized matrix is imported into the MRCNN model for feature extraction, which possesses a multiscale hopping structure allowing the model to extract more local and global features. Consequently, the network of the MRCNN model is easier for constructing more reasonable health indicators by training the model with excellent performance. To highlight the degradation trend, the constructed health indicators are

smoothed by utilizing the exponential moving average (EMA) algorithm. Finally, linear regression is exploited to predict the remaining useful life.

The main contributions of this paper can be summarized as follows:

- (1) The proposed MRCNN model especially constructs a multiscale hopping structure, which accurately predicts the RUL of the bearings
- (2) The MRCNN model utilizes global average pooling instead of flatten layer to reduce the number of parameters, shorten the training time, and reduce the problem of overfitting
- (3) The obtained HI is smoothed in the final stage by using an exponential moving average algorithm, which highlights the recessionary trend and makes the predicted RUL more accurate

The remainder of this paper is organized as follows: Section 2 provides the details of the materials and methods of the MRCNN model. In Section 3, the specific experimental steps to verify the effectiveness of the MRCNN model using the PRONOSTIA dataset are introduced. The experimental

results are compared with existing data-driven algorithms in terms of prediction accuracy. Finally, Section 4 concludes this paper.

2. Materials and Methods

2.1. Data Analysis. In this paper, the famous public dataset PRONOSTIA at the IEEE PHM 2012 Data Challenge is utilized to validate the accuracy of the MRCNN model while predicting bearing RUL. As shown in Figure 1, the sensors on the platform collect vibration acceleration signals (horizontal vibration acceleration signals and vertical vibration acceleration signals) during bearing running. Consequently, the MRCNN model utilizes horizontal and vertical vibration acceleration signals datasets in this paper. The experimental platform tests a total of 17 bearings for degradation experiments, which are divided into three types according to the external forces and rotational speeds. Each type has two training datasets, including vibration signals with a complete life cycle, and the test set is truncated (the first part of the life cycle) vibration signals. Specifically, the first two types have the vibration signals of two bearings as the training dataset and five bearings as the test dataset (condition 1 and condition 2 in Table 1), and the third type has the vibration signals of two bearings as the training dataset and only one bearing as the test dataset (condition 3 in Table 1). Specifically, vibration signal data is collected by the sensor at 10 s intervals, with a collection duration of 0.1 s and a sampling frequency of 25600 Hz. Since this paper mainly studies the prediction of bearing degradation trends under normal use, the first working condition is the most eligible. Consequently, the experiment mainly utilizes the data under the first working condition. The other two working conditions exert too much external force and the unpredictable events of the bearings increase; thus, condition 2 and condition 3 can simulate the bearing degradation prediction under sudden events, which is challenging. The specific experimental data are shown in Table 1.

Through analyzing the vibration signals of Bearing1_1 and Bearing1_2, Figure 2 illustrates that accurately predicting the RUL of the bearings is unrealistic only by the amplitude of the vibration signal. Fortunately, the CWT is an effective signal processing method, which extracts time-frequency features by transforming the original time-domain vibration signal into a 2-D time-frequency matrix. The CWT algorithm can effectively demonstrate the different changes in the frequency domain as time grows. For a stationary signal, its time-domain characteristics can well describe its relative variation. As seen from Figure 2 (Bearing1_2), the vibration signals of the bearings are nonstationary and have a weak fault signal in the strong noise background. Time-frequency analysis is an effective method for analyzing nonstationary and transient vibration signals. This paper utilizes the CWT to generate the time-frequency matrix as the dataset for training, while other papers use time-frequency images as the dataset, but this operation will encounter two problems of uniform colour bars and loss of accuracy. Consequently, as shown in Figures 3 and 4, time-frequency diagrams are only used as

TABLE 1: Experimental data specification.

	Condition 1	Condition 2	Condition 3
Load (N)	4000	4200	5000
Speed (rpm)	1800	1650	1500
Training dataset	Bearing1_1	Bearing2_1	Bearing3_1
	Bearing1_2	Bearing2_2	Bearing3_2
	Bearing1_3	Bearing2_3	
	Bearing1_4	Bearing2_4	
Testing dataset	Bearing1_5	Bearing2_5	Bearing3_3
	Bearing1_6	Bearing2_6	
	Bearing1_7	Bearing2_7	

a demonstration of time-frequency characteristics. Figure 3 shows the waveform and time-frequency images of the first and last samples of Bearing1_1. According to the increase of the high-frequency component in the time-frequency plot, the colour change becomes increasingly obvious and the degradation characteristics are clearly demonstrated. In order to show the degradation trend of bearings completely, Figure 4 shows the time-frequency diagrams of Bearing1_1 and Bearing1_2 with the display step of 25 percent of the complete life. Obviously, the bearing failure becomes more serious, and the colour change on the time-frequency diagram becomes more obvious with time. This shows that time-frequency analysis can well expose the fault characteristics of bearings.

2.2. Proposed Framework. Model-based methods and common data-driven methods have the drawback of low accuracy in predicting RUL. Consequently, this paper proposes a more accurate RUL prediction method based on CWT and MRCNN models. The framework diagram of the proposed method is shown in Figure 5 as follows:

In this paper, the proposed prediction algorithm is generally divided into three stages: the first stage is feature extraction, where the vibration signal is converted into a 2-D time-frequency matrix by CWT to extract time-frequency features. Concurrently, the training dataset and test dataset are obtained by extracting the time-frequency features utilizing the CWT algorithm. In order to cut down the input volume of the model, the size of the 2-D matrix is reduced by utilizing the nearest interpolation algorithm. The second stage is the offline health indicators construction, where the resized time-frequency matrix is imported into MRCNN. Since CNN is a supervised learning method, the design of its labels largely affects its RUL prediction results. This paper assumes that the bearing degradation trend is committed to a linear trend, thus setting the labels to decrease linearly from 1 to 0 to learn each stage's degradation characteristics. HIs are structured by importing the test dataset into the trained MRCNN model. Finally, the third stage is online RUL prediction. To highlight the degradation trend, the structured HIs are smoothed by an exponential moving average algorithm. The RUL was finally obtained by linear regression. The main techniques are detailed described below.

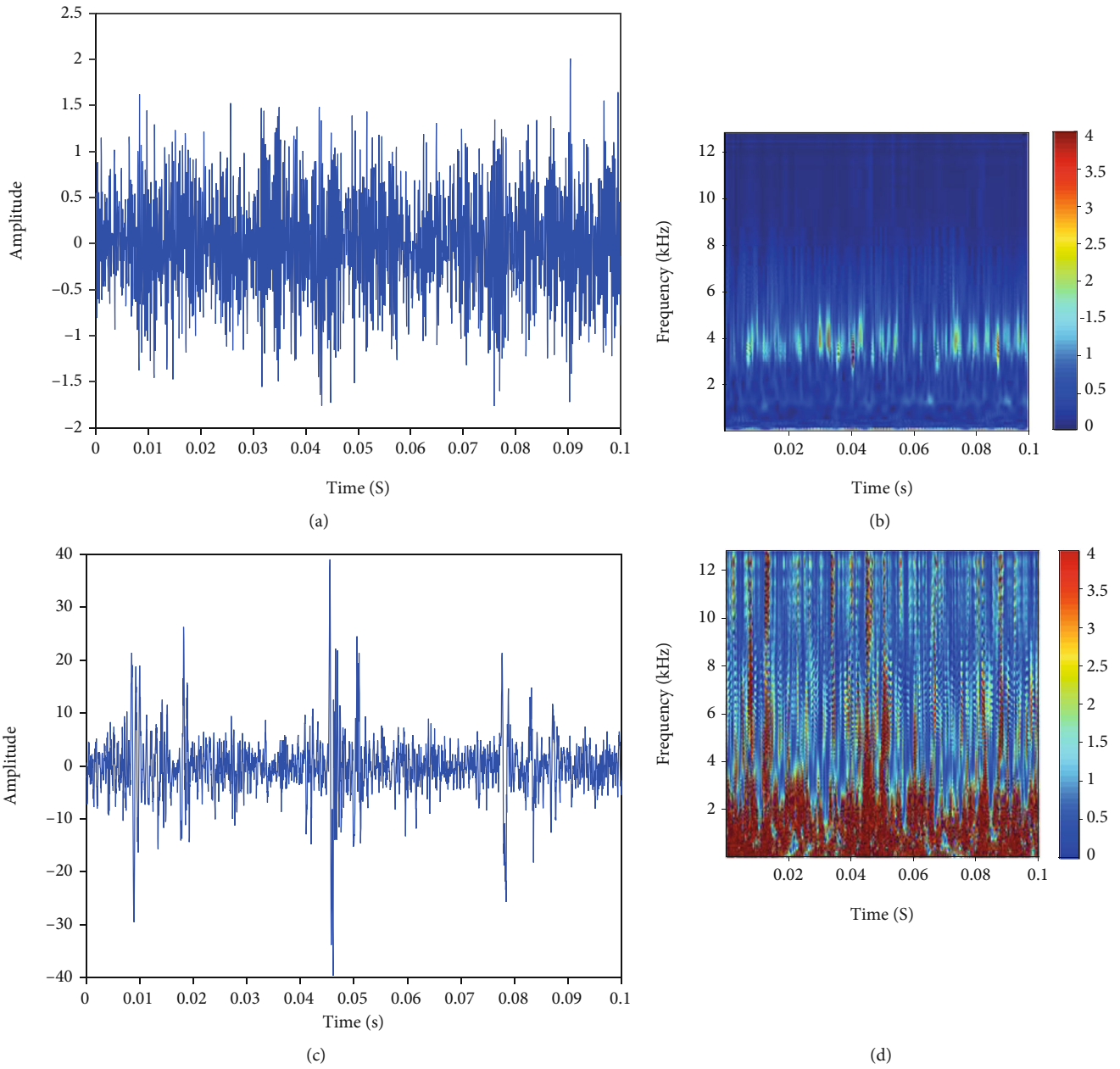


FIGURE 3: Bearing1_1: (a) waveform for the first sample; (b) time-frequency image for the first sample; (c) waveform for the last sample; and (d) time-frequency diagram for the last sample.

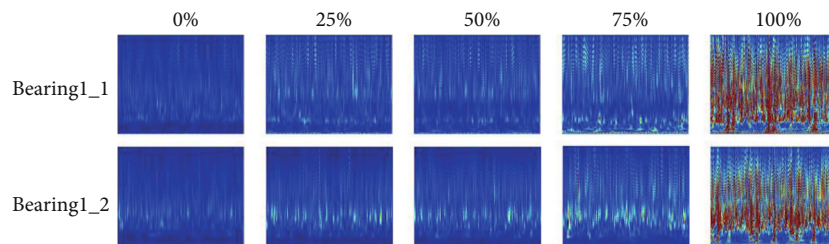


FIGURE 4: Bearing1_1 and Bearing1_2 display time-frequency images every 25% in the whole life.

2.3. *Continuous Wavelet Transform.* With the development of signal analysis techniques, different time-frequency transform techniques have emerged accordingly. Fourier trans-

form, as the most basic time-frequency transform method, can transform the signal from the time domain to the frequency domain. Consequently, the Fourier transform can

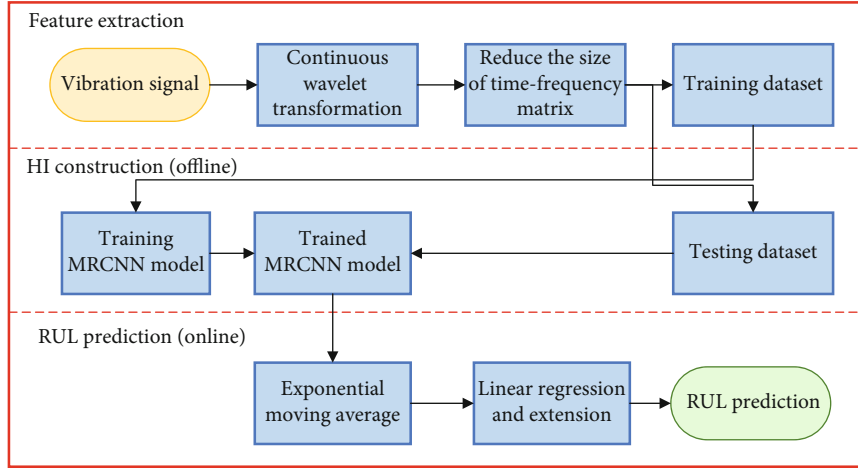


FIGURE 5: Framework of the proposed method.

extract much information about the essence of the signal. However, the Fourier transform has the shortcoming in reflecting the trend of frequency variation over time. The windowed Fourier transform comes out accordingly, which enables us to observe the frequency components on different time domains by choosing the corresponding windows. But the fixed-size window makes us feel inconvenient. Consequently, the continuous wavelet transform was invented to solve this problem. The continuous wavelet transform replaces the infinite length trigonometric basis with a finite length decaying wavelet basis, which makes us intuitively feel the changes of time-frequency characteristics [21]. The vibration signal of the faulty bearing contains periodic pulses whose shape is similar to Morlet wavelets; thus, the experiment in this paper utilizes the Morlet wavelet basis function to transform the original vibration signal. The Morlet wavelet basis function is shown as follows:

$$\psi(t) = \exp(i\omega_0 t) \exp\left(-\frac{t^2}{2}\right), \quad (1)$$

where $\exp(i\omega_0 t)$ is called the complex trigonometric function whose role is to identify the frequency and $\exp(-t^2/2)$ is called the decay function, which can guarantee finite support in its time domain. In order to get different frequency signals, the original signals are multiplied by different scales of wavelet basis functions $\psi(t)$. Consequently, the different frequency signals can be obtained only by stretching the wavelet basis function. The continuous wavelet transform formula is as follows:

$$\begin{aligned} U(\alpha, \beta) &= \langle x(t), \psi_{\alpha\beta} \rangle = \int_{-\infty}^{\infty} x(t) \bar{\psi}_{\alpha\beta}(t) dt \\ &= \int_{-\infty}^{\infty} (t) \frac{1}{\sqrt{\alpha}} \bar{\psi}\left(\frac{t-\beta}{\alpha}\right) dt, \end{aligned} \quad (2)$$

where $x(t)$ is the input original 1-D signal and $\psi(t)$ is the mother wavelet function, $\bar{\psi}_{\alpha\beta}(t)$ is the complex conjugate function of $\psi_{\alpha\beta}(t)$, and the coefficients α, β , respectively,

represent the scale coefficients (frequency variables) and translation coefficients (time variables). By setting the input 1-D signal $x(t)$ and the mother wavelet function $\psi(t)$, the time-frequency coefficient matrix can be obtained according to the above integral (formula (2)). Consequently, the CWT can identify a specific frequency β under a specific time α .

2.4. Multiscale Residual Convolutional Neural Network. CNN is a deep learning model inspired by the visual cortex of the brain [22], which extracts the location information of the original image by convolutional operations. Since the perceptual fields of convolution kernels with different sizes on the input image are different, the learned features are also different. Consequently, the multiscale convolution structure contributes to the network learning more local and global features, which has a good application in the field of RUL prediction [23]. CNN mainly consists of three structures: convolutional layer, pooling layer, and fully connected layer. Deep CNN achieves stronger data representation capability by stacking convolution and pooling layers, whose end generally adopts a fully connected layer for classification and regression. Convolution kernels, also known as weights, are slid over the input image by performing convolution operations on each corresponding local perceptual field. The weight sharing greatly reduces the number of parameters and avoids the overfitting problem. The input image is convolved with a convolution layer to obtain the feature map, which is a feature representation of the image, and its calculation formula is as follows:

$$x_j^t = \varphi\left(\sum_i x_i^{t-1} * k_{ij}^t + b_j^t\right), \quad (3)$$

where x_j^t is the feature map of the t -th layer and the j -th output, k_{ij}^t is the convolution kernel of the t -th layer, b_j^t is the bias of the t -th layer, and φ is the nonlinear activation function. This means that all feature maps in layer $t-1$ do the convolution operation with the convolution kernel in layer t , and the convolution result is summed with the bias b_j^t . Finally, the feature map in layer t is obtained by the nonlinear activation function.

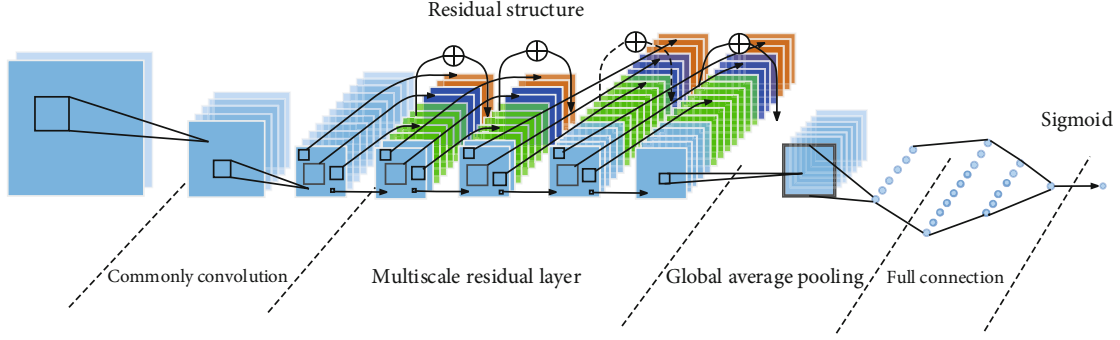


FIGURE 6: MRCNN structure.

A pooling layer is usually added between adjacent convolutional layers. The pooling operation can reduce the size of the parameter matrix, simplify the complexity of the network, speed up the computation, and prevent overfitting. The output feature map of the pooling layer has the same variance as the input feature map. As shown in Figure 6, MRCNN model utilizes two pooling methods: maximum pooling and global average pooling. Maximum pooling takes the maximum value of the data within the perceptual field to achieve the effect of subsampling. The global average pooling layer takes the average of the entire feature map. In the MRCNN model, global average pooling is utilized instead of flatten layers to reduce the number of parameters, shorten the training time, and reduce the occurrence of overfitting problems.

The first two layers of the MRCNN model are ordinary convolutional layers, which consist of convolution, BN (batch normalization), Relu activation function, and max pooling. The Relu activation function is formulated as $\text{Relu}(x) = \max(0, x)$, which serves to enable the network to perform nonlinear operations and to keep the gradient from decaying when the input x is greater than 0, alleviating the vanishing gradient problem. Since BN operation can maintain the input data in the activation function sensitive region (mean γ and variance β learned by BN), speed up the convergence speed during model training, and make the model training process more stable. Consequently, adding the BN operation before the activation function is necessary. The BN formula is as follows:

$$y_i^{(b)} = \text{BN}(x_i)^{(b)} = \gamma \cdot \left(\frac{x_i^{(b)} - \mu(x_i)}{\sqrt{\sigma(x_i)^2 + \mathcal{E}}} \right) + \beta, \quad (4)$$

where x_i is a row vector of size one batch, which represents the value of the b -th input node of the layer at the i -th sample in the current batch. Where μ and σ , respectively, represent the mean and variance of the row, and a minimal amount \mathcal{E} is introduced to prevent the denominator from being zero. The purpose of the BN operation is to obtain a data distribution that makes convergence faster during training $y_i^{(b)}$. The training learning parameters γ and β , respectively, represent the mean and variance of the data after BN operation.

In the MRCNN model, from the third layer to the sixth layer is the multiscale residual layer composed of four multiscale residual blocks. The specific multiscale residual block structure is shown in Figure 7. The feature map of the upper layer is input into the multiscale residual block, which experiences four structures: the first structure has a convolution layer with a kernel size of 1×1 , the second structure has two convolution layers with kernel sizes of 3×3 and 1×1 , respectively, the third structure has both convolution and pooling layers, in which the convolution kernel size is 1×1 , and the pooling layer kernel size is 3×3 . The fourth structure has three convolutional layers, whose first two layers have a kernel size of 3×3 and whose last layer has a kernel size of 1×1 . The purpose of adopting the above structure is to utilize multiscale convolution kernels with different perceptual fields on the upper feature map, which can extract a variety of different features. Since the convolutional layer with a kernel size of 3×3 has a perceptual field of 9 on the previous feature map, the superposition of two layers with a kernel size of 3×3 has a perceptual field of 25 on the previous feature map at a step size of 1. Consequently, the superposition of two layers with a kernel size of 3×3 can replace a convolutional layer with a kernel size of 5×5 , and more information can be learned because of the deeper network. The maximum pooling layer can learn the edge and texture structure of the image. The 1×1 convolutional layer in the multiscale residual block can be seen as a fully connected layer, which can serve to change the number of output channels. Eventually, these four structures are stitched together. The residual structure of the multiscale residual block is learned from the ResNet [23] network, which solves the network degradation and deep network model vanishing gradient and exploding gradient problems. Since the residual structure learns the difference between the target value and the input value when training to the optimal model, the structure will train the redundant layers into the identity function. Therefore, when the number of network layers deepens, its accuracy will not decline. In the fifth layer of the MRCNN model, due to the change in the number of channels, the residual structure is represented by a dashed line. The multiscale convolution block is activated by Relu nonlinear unit. Finally, the MRCNN model adopts the full connection layer and sigmoid activation function to distribute the output HI in the region of 0 to 1.

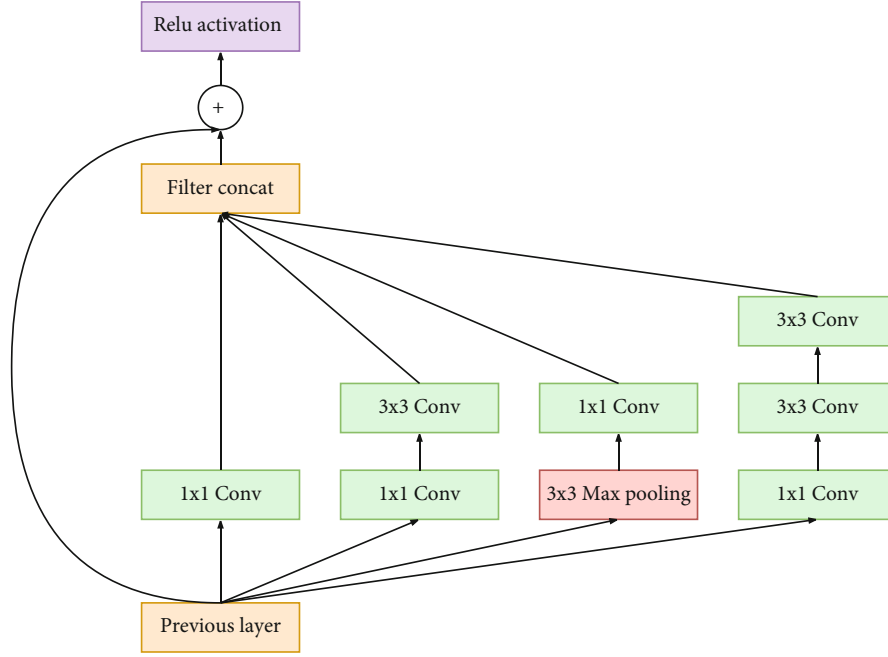


FIGURE 7: Multiscale residual block.

2.5. Exponential Moving Average. The HI generated by the MRCNN model is smoothed by exploiting an exponential moving average algorithm. Although the HI shows a clear degradation trend from the overall, there will exist some HI deviation from the overall trend due to the interference of some strong noise. Consequently, in this paper, HI is discrete in narrowing the specified period. The exponential moving average algorithm is utilized to perform the smoothing operation on HI. The exponential moving average equation is presented as follows:

$$v_t = \beta v_{t-1} + (1 - \beta)\theta_t, \quad (5)$$

where v_t denotes the current predicted value, v_{t-1} is the previous moment predicted value, θ_t refers to the current HI, and β is the decay factor. According to Equation (5), the current prediction value v_t is obtained by weighting the previous prediction value v_{t-1} with the current θ_t . Therefore, Equation (5) can be expanded as follows:

$$v_t = (1 - \beta)\theta_t + \beta(1 - \beta)\theta_{t-1} + \beta^2(1 - \beta)\theta_{t-2} + \dots + \beta^{t-1}(1 - \beta)\theta_1. \quad (6)$$

As seen from Equation (6), the current prediction value v_t is associated with the values of all previous moments, where the last predicted value v_t is obtained by a process called sliding backward from the first value θ_1 to the last value θ_t . As a result, the relationship between the current predicted value v_t and the previous value θ_n shows $\beta^n(1 - \beta)$ exponential decay. Ultimately, the exponential moving average algorithm makes the recession trend more visible and enables a more accurate prediction of RUL.

2.6. Linear Regression Prediction. The final step of RUL prediction is a linear regression operation on the smoothed health indicators. Since the label design of MRCNN model is that the bearing decay follows a linear change, the label design is to reduce linearly from 1 to 0 to learn the degradation characteristics of each stage. Consequently, for accurate prediction of RUL, a linear regression prediction in the last step is essential. Formally, the linear equation $y = at + b$ is exploited to fit the last 1/3 HI values of the life cycle of the bearings, and the final RUL is obtained by extension. Referring to the ordinary least squares, the slope a and the intercept b can be obtained by the following two formulas [24].

$$a = \frac{\sum_{i=0}^N t_i y_i - (N + 1)\bar{t}\bar{y}}{\sum_{i=0}^N t_i^2 - (N + 1)\bar{t}^2}, \quad (7)$$

$$b = \bar{y} - a\bar{t},$$

where t_i and y_i represent the time of i -th and its corresponding predicted HI. The final predicted RUL (\widehat{RUL}) can be obtained by intercepting b over the upper slope a and subtracting the truncated time point t_N , which is shown as follows:

$$\widehat{RUL} = \frac{b}{a} - t_N. \quad (8)$$

3. Results and Discussion

3.1. MRCNN Model Construction HI. In this section, the structure of the MRCNN model and the experimental procedure of constructing HI is described in detail. Firstly, the original vibration signals are converted into a 2-D matrix of size 2560×2560 by CWT. In order to cut down the input

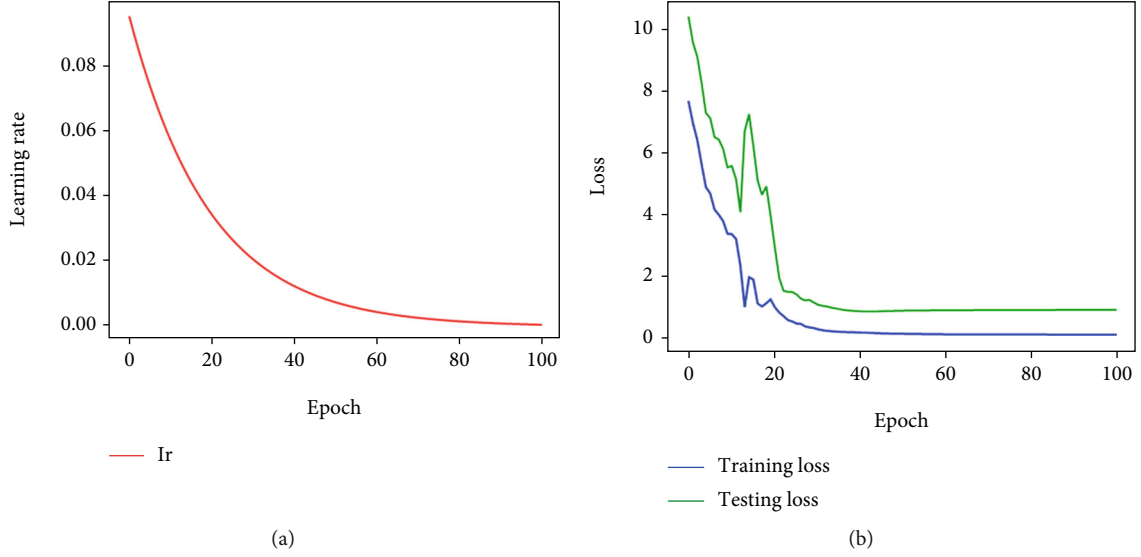


FIGURE 8: (a) Learning rate. (b) Training loss and testing loss.

volume of the model, the size of the 2-D matrix is reduced to 224×224 by utilizing the nearest interpolation algorithm, which greatly saves massive amounts of training time. The PRONOSTIA public dataset collects vibration signals in horizontal and vertical directions. Consequently, the input of MRCNN model is set as two channel numbers, which, respectively, correspond to the time-frequency matrices in these two directions. Afterward, the first two layers of MRCNN model are ordinary convolution layers, which utilize the structures of convolution, BN, Relu, and maximum pool. The convolution kernel size of the first layer is 7×7 and the step size is 1. The first layer convolution operation will not change the size of the 2-D matrix. The 2-D matrix output from the first layer of MRCNN model is halved to the size of 112×112 , and the number of output channels is 64. The second convolutional layer uses a similar structure, doubling the number of output channels to 128×128 and halving the 2-D matrix size to 56×56 . The difference is that the convolution operation has a convolution kernel of 3×3 , stride of 1 and padding of 1. Next is the multiscale residual layer, which is composed of four multiscale residual blocks, each of which has four multiscale feature mapping operations. Specifically, the first operation is 1×1 convolution to extract feature map. The second is 1×1 convolution plus 3×3 convolution, and the output ratio is 1:2. The third is max pooling plus 1×1 convolution, and the output ratio is 1:1. The fourth is 1×1 convolution plus two 3×3 convolutions, and the output ratio is 1:2:2. Overall, the output ratio of these four channels is 2:4:1:1. The above parameters are obtained through massive amounts of experimental tuning parameters, which have strong applicability. In particular, MRCNN uses the residual hopping structure from layer 3 to layer 6 but further doubles the number of channels from layer 4 to layer 5 to 256×256 . Therefore, the residual hopping structure here is first realized by changing the number of channels through 1×1 convolution and then hopping. The seventh layer of MRCNN is a normal convolutional layer with a convolutional kernel size of 3×3 , a stride of 1,

a padding of 1, and an output channel of 64. The eighth layer of MRCNN is the global averaging pooling layer, which is implemented by taking the average of a 56×56 size 2-D matrix and outputting 64 parameters. The ninth to eleventh layers of MRCNN are full connection layers, and the number of neurons in each layer is 128, 64, and 1, respectively. Finally, the MRCNN model obtains HIs through sigmoid activation function, which makes the distribution of HIs from 0 to 1.

After CWT algorithm transformation, the training datasets from PRONOSTIA public dataset are successively imported into MRCNN model for training, because an obvious advantage of data-driven methods is that it does not require massive amounts of a priori knowledge. The label is set to decrease linearly from 1 to 0. In this paper, the full life of the bearing is recorded as T_a and the current operating time is recorded as T_c . The true label of HI can be obtained by the following formula:

$$\text{label} = 1 - \frac{T_c}{T_a}. \quad (9)$$

The HI represents the health of bearings, 1 indicates complete health, and 0 indicates damage and no longer usable. The loss function of the MRCNN model utilizes mean square error (MSE), which is widely adopted as the loss function of regression model. MSE is shown as follows:

$$L(\hat{y}; y) = \frac{1}{2} \|\hat{y} - y\|^2, \quad (10)$$

where \hat{y} is the predicted value of the MRCNN model and y is the true value. The MRCNN model adjusts the parameters by back propagation and gradient descent to reduce the loss function and make the target value closer to the true value. Since the optimizer utilizes stochastic gradient descent (SGD), which approximates the gradient on the loss function of the entire dataset by using the gradient on the loss

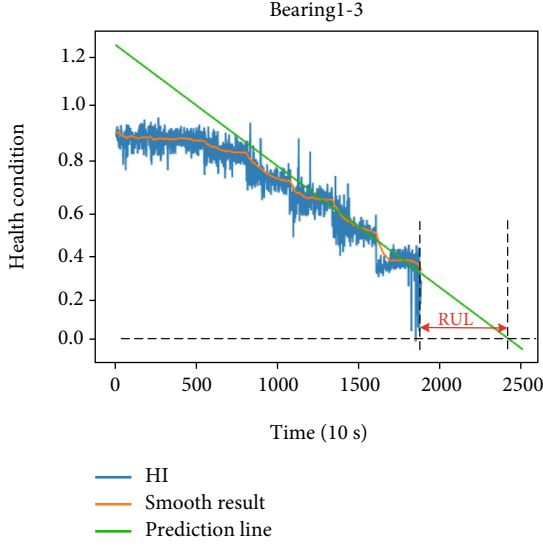


FIGURE 9: RUL prediction result of Bearing1_3.

function of small batch data, therefore, it is easier to adjust the direction of gradient descent and achieve global optimization in the training process. The learning rate is adjusted by the exponential decay method, whose initial learning rate is 0.1 and the multiplication factor is set to 0.95. Dynamically, decaying the learning rate can quickly reduce the loss in the early iteration stage and slightly reduce the loss in the later stage. The learning rate decay is shown as follows:

$$lr = lr \cdot (0.95)^n, \quad (11)$$

where lr denotes the learning rate and n is the number of iterative rounds. The training dataset includes Bearing1_1 and Bearing1_2, and the test dataset includes Bearing1_3 to Bearing1_7. In this paper, a total of 100 training rounds were performed. Figure 8(a) illustrates that the learning rate decays exponentially with the increase of round, and the decline law of the loss function is the same as expected. In the early stage, the loss is rapidly reduced, and in the later stage, the loss is slightly reduced. However, as shown in Figure 8(b), when the 5th to 10th rounds of training are performed, the loss function on the training and test dataset increases instead, which is caused by the excessive learning rate at that time. By utilizing the decay operation of the learning rate, the loss can be correctly reduced in the subsequent rounds. In addition, the MRCNN model adopts global average pooling before the fully connected layer to reduce the problem of overfitting.

The trained MRCNN model is used to test Bearing1_3. The GPU model used in the experimental platform is GTX1080ti, and the MRCNN model is implemented based on the deep learning framework Pytorch. The truncation time point of test dataset Bearing1_3 is 18010s, and it spends 6.4s for MRCNN model to construct HIs. The MRCNN model constructs HIs on the test dataset Bearing1_3, whose results are shown in the blue curve in Figure 9. The degradation trend of bearings can be clearly

seen from HI in Figure 9. Firstly, in the range from 0s to 7500s, the bearing degradation is slow, and the bearing can operate stably in the early stage of its service life. Afterward, the degradation trend suddenly becomes faster at 7500s, 10500s, 13000s, 15700s, and 17500s. Due to the influence of external force, the degradation of bearings gradually accelerates until it is finally damaged. Although the HIs shows a clear degradation trend from the overall, there will exist some HI deviation from the overall trend due to the interference of some strong noise. Consequently, as shown in Figure 9, HI is discrete in narrowing the specified period.

3.2. RUL Prediction Results. After obtaining the HI through the MRCNN model, the next step is to perform RUL prediction based on the HIs. Since the obtained HI distribution is more discrete, this problem is solved by fitting HIs with an exponential sliding average algorithm. According to formulas (5) and (6), the current predicted value is associated with the values of all previous moments. The last predicted value is obtained by a process like sliding backwards in sequence from the first value to the last value, doing exponential decay according to the decay factor. The decay factor in the experiment takes the value of 0.95. Consequently, the exponential sliding average algorithm makes the decline trend more obvious and enables more accurate prediction of the RUL, as shown in the orange curve in Figure 9. In this paper, through massive amounts of experiments, the following operations are the most accurate prediction of RUL: take the last 30% of the smooth data and expand it through linear fitting. As shown by the green line in Figure 9, the predicted life can be obtained by intersecting with the line whose ordinate is equal to zero. Consequently, the RUL of bearing can be obtained by subtracting the current time from the predicted life. The results of RUL prediction from Bearing1_4 to Bearing1_7 are shown in Figure 10. Specifically, Bearing1_4 predicts a full life of 14740s, Bearing1_5 predicts a full life of 24980s, Bearing1_6 predicts a full life of 24300s, and Bearing1_7 predicts a full life of 21880s.

Regarding the performance evaluation metrics of RUL prediction, this paper adopts the method given in the IEEE PHM 2012 Prognostic Challenge: error percentage, evaluation metric A_i , and mean value of evaluation metric A_i . The error percentage is shown in

$$Er_i = 100 \times \frac{\text{ActRUL}_i - \widehat{\text{RUL}}_i}{\text{ActRUL}_i}, \quad (12)$$

where ActRUL_i represents the true RUL of the i -th bearing and $\widehat{\text{RUL}}_i$ represents the predicted RUL of the i -th bearing, since the error percentage Er represents the deviation of the predicted RUL from the true RUL. Consequently, the closer the predicted RUL is to the true RUL, the closer the error percentage is to 0. In addition, the evaluation index is shown in

$$A_i \begin{cases} \exp(-\ln(0.5) \cdot (Er_i/5)) & \text{if } Er_i \leq 0, \\ \exp(\ln(0.5) \cdot (Er_i/20)) & \text{if } Er_i > 0. \end{cases} \quad (13)$$

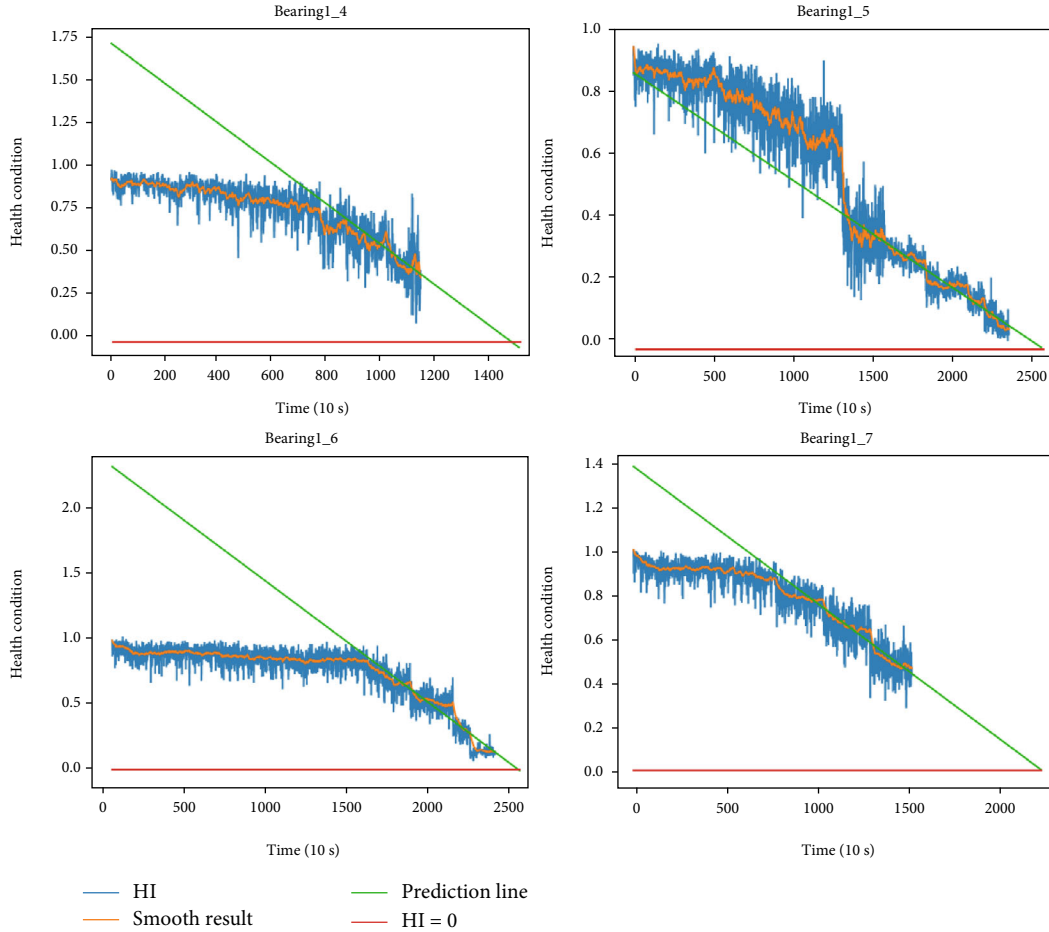


FIGURE 10: RUL prediction result of Bearing1_3 to Bearing1_7.

TABLE 2: Comparison results of this research and related researches in the PRONOSTIA dataset.

Testing dataset	Current time (s)	Actual RUL (s)	Predicted RUL (s)	Er_1	Er_2 [24]	Er_3 [5]	Er_4 [25]
Bearing1_3	18010	5730	5470	4.53	37	-0.35	7.62
Bearing1_4	11380	2890	3360	-16.26	80	5.60	-157.71
Bearing1_5	23010	1610	1970	-22.36	9	100.00	-72.57
Bearing1_6	23010	1460	1290	11.64	-5	28.08	0.93
Bearing1_7	15010	7570	6870	9.24	-2	19.55	85.99
$ \overline{Er} $				12.81	26.6	30.71	64.96
Score				0.47	0.46	0.45	0.35

The evaluation metric A_i represents a measure of RUL predictive performance. If the Er_i is 0, A_i is equal to 1. As the absolute value of Er_i increases, the A_i becomes larger. However, the cost of overpredicting and underpredicting the RUL is different. In the field of RUL prediction, the cost of underestimating RUL is lower than overestimating RUL. Equation (13) captures this well. The final score is shown in Equation (14), which represents the mean value of the evaluation metrics.

$$\text{Score} = \frac{1}{n} \sum_{i=1}^n A_i. \quad (14)$$

To verify the superiority of the method proposed in this paper, other data-driven methods using the same dataset are compared. The literature [24] is the champion model of IEEE 2012 Data Challenge. Lei et al. [5] proposed WMQE model, which has good prediction performance by fusing several weighted features to predict RUL through correlation clustering between 28 features of bearings. Chen Y et al. [25] extracted five bandpass energy values of the spectrum as features to construct a network based on code and decoding framework and attention mechanism for network prediction of RUL. Attention model based on signal decomposition brings us a new idea. Table 2 writes the final scores for each bearing current time, actual RUL, predicted RUL by this

method, and the percentage of error compared to the other three methods. In addition, the absolute mean of the error percentage $|\overline{Er}|$ is added in this paper, which judges the merits of the method by comparing the dispersion degree of the RUL prediction error. As seen from Table 2, $|\overline{Er}|$ is 12.81 and the final score is 0.47. Both in terms of prediction accuracy and dispersion of prediction, the method in this paper is better than the other three methods.

3.3. Discussion. The effectiveness and advantage of the proposed MRCNN method have been verified by experimental data and point out the shortcomings of the MRCNN model and the direction of improvement in the future.

- (1) The MRCNN model adopts the time-frequency matrix transformed from the vibration signal, which saves the training time by reducing its size. However, reducing the size of features will lose a lot of useful information. Therefore, it is a very important problem to achieve a balance between the cost of time and the input of enough features
- (2) The structure of MRCNN model design is complex, and the optimal parameters are obtained through a large number of experiments. In the future, attention model will be used, which can enable neural network to learn more important information of the channels
- (3) In addition to time and frequency features, other useful features can be used, such as temperature and humidity. If there is a suitable dataset, we will use these features

4. Conclusions

In this paper, the CWT algorithm is utilized to convert the time-domain signal into a time-frequency domain 2-D matrix firstly. Afterward, HIs are successfully constructed by MRCNN model. The advantages of the MRCNN model are summarized as follows. The multiscale structure of MRCNN model can extract the local and global features of multiple scales. The residual structure of MRCNN model can make the network easier to train into an accurate model. The MRCNN model uses global average pooling instead of flatten layers to reduce the problem of overfitting. Concurrently, the EMA algorithm is used to highlight the degeneration trend. RUL was finally obtained by linear regression prediction. The RUL prediction results prove the validity of the method. Finally, we compare with the method in recent years that predicts RUL on the same dataset, and the results show that the method in this paper has the best prediction results in terms of prediction accuracy and dispersion of prediction.

Data Availability

In this paper, the public dataset PRONOSTIA at the IEEE PHM 2012 Data Challenge can be obtained through the link: <https://github.com/wkzs111/phm-ieee-2012-data-challenge-dataset>.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the State Key Laboratory of Process Automation in Mining & Metallurgy and Beijing Key Laboratory of Process Automation in Mining & Metallurgy under the Grant No. BGRIMM-KZSKL-2020-02. This research was partly supported by PADA.

References

- [1] C. H. Hu, H. Pei, X. S. Si, D. B. du, Z. N. Pang, and X. Wang, "A prognostic model based on DBN and diffusion process for degrading bearing," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 10, pp. 8767–8777, 2020.
- [2] M. Rezamand, M. Kordestani, M. E. Orchard, R. Carriveau, D. S. K. Ting, and M. Saif, "Improved remaining useful life estimation of wind turbine drivetrain bearings under varying operating conditions," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 1742–1752, 2021.
- [3] F. Shen and R. Yan, "A new intermediate domain SVM-based transfer model for rolling bearing RUL prediction," *IEEE/ASME Transactions on Mechatronics*, vol. 99, p. 1, 2021.
- [4] M. Elforjani and S. Shanbr, "Prognosis of bearing acoustic emission signals using supervised machine learning," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5864–5871, 2018.
- [5] Y. Lei, N. Li, S. Gontarz, J. Lin, S. Radkowski, and J. Dybala, "A model-based method for remaining useful life prediction of machinery," *IEEE Transactions on Reliability*, vol. 65, no. 3, pp. 1314–1326, 2016.
- [6] B. Wang, Y. Lei, N. Li, and W. Wang, "Multiscale convolutional attention network for predicting remaining useful life of machinery," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 8, pp. 7496–7504, 2021.
- [7] Q. Zhai and Z. S. Ye, "RUL prediction of deteriorating products using an adaptive Wiener process model," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 6, pp. 2911–2921, 2017.
- [8] N. Li, Y. Lei, N. Gebraeel et al., "Multi-sensor data-driven remaining useful life prediction of semi-observable systems," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 11, pp. 11482–11491, 2021.
- [9] Y. Song, S. Gao, Y. Li, L. Jia, Q. Li, and F. Pang, "Distributed attention-based temporal convolutional network for remaining useful life prediction," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9594–9602, 2021.
- [10] Y.-S. Shih and J.-J. Chen, "Analysis of fatigue crack growth on a cracked shaft," *International Journal of Fatigue*, vol. 19, no. 6, pp. 477–486, 1997.
- [11] Y. Li, S. Billington, C. Zhang, T. Kurfess, S. Danyluk, and S. Liang, "Adaptive prognostics for rolling element bearing condition," *Mechanical Systems and Signal Processing*, vol. 13, no. 1, pp. 103–113, 1999.
- [12] K. He, Z. Su, X. Tian, H. Yu, and M. Luo, "RUL prediction of wind turbine gearbox bearings based on self-calibration temporal convolutional network," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, 2022.

- [13] X. Yang, Y. Zheng, Y. Zhang, D. S. H. Wong, and W. Yang, "Bearing remaining useful life prediction based on regression Shapaleet and graph neural network," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–12, 2022.
- [14] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: a systematic review from data acquisition to RUL prediction," *Mechanical Systems and Signal Processing*, vol. 104, pp. 799–834, 2018.
- [15] N. Gebraeel, M. Lawley, R. Liu, and V. Parmeshwaran, "Residual life predictions from vibration-based degradation signals: a neural network approach," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 3, pp. 694–700, 2004.
- [16] R. Zhao, R. Yan, J. Wang, and K. Mao, "Learning to monitor machine health with convolutional bi-directional LSTM networks," *Sensors*, vol. 17, no. 2, p. 273, 2017.
- [17] Z. Huijie, R. Ting, W. Xinqing, Z. You, and F. Husheng, "Fault diagnosis of hydraulic pump based on stacked autoencoders," in *2015 12th IEEE International Conference on Electronic Measurement & Instruments (ICEMI)* pp. 58–62, Qingdao, China, 2015.
- [18] A. Malhi, R. Yan, and R. X. Gao, "Prognosis of defect propagation based on recurrent neural networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 3, pp. 703–711, 2011.
- [19] W. Peng, Z. S. Ye, and N. Chen, "Bayesian deep-learning-based health prognostics toward prognostics uncertainty," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 3, pp. 2283–2293, 2020.
- [20] P. Nectoux, R. Gouriveau, K. Medjaher et al., "Pronostia: an experimental platform for bearings accelerated degradation tests," in *IEEE International Conference on Prognostics and Health Management, PHM'12* pp. 1–8, Denver, Colorado, United States, 2012.
- [21] J. Zhu, N. Chen, and W. Peng, "Estimation of bearing remaining useful life based on multiscale convolutional neural network," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 4, pp. 3208–3216, 2019.
- [22] J. Wang, J. Zhuang, L. Duan, and W. Cheng, "A multi-scale convolution neural network for featureless fault diagnosis," in *Proceedings of the 2016 International Symposium on Flexible Automation (ISFA)*, pp. 65–70, Cleveland, OH, USA, August 2016.
- [23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* pp. 770–778, Las Vegas, NV, USA, 2016.
- [24] E. Sutrisno, H. Oh, A. S. S. Vasan, and M. Pecht, "Estimation of Remaining Useful Life of Ball Bearings Using Data Driven Methodologies," in *Proceedings of the 2012 IEEE Conference on Prognostics and Health Management*, pp. 1–7, Denver, CO, USA, June 2012.
- [25] Y. Chen, G. Peng, Z. Zhu, and S. Li, "A novel deep learning method based on attention mechanism for bearing remaining useful life prediction," *Applied Soft Computing*, vol. 86, article 105919, 2020.