Remaining Useful Life Prediction of Rolling Bearings Based on Adaptive Continuous Deep Belief Networks and Improved Kernel Extreme Learning Machine

Meng Zhou[1] | Jing Wang[1] | Yuntao Shi[1] | Zhenhua Wang[2] | Vicenç Puig[3]

[1] School of Electrical and Control Engineering, North China University of Technology, Beijing, China | [2]School of Astronautics, Harbin Institute of Technology, Harbin, China | [3]Advanced Control Systems Research Group at Institutde Robòtical, CSIC-UPC, Universitat Politècnica de Catalunya-Barcelona Tech, Barcelone, Spain

Correspondence: Jing Wang (jwang@ncut.edu.cn)

Funding: This article is funded by National Natural Science Foundation of China (62273007, 62373005); the Project of Cultivation for Young Top-notch Talents of Beijing Municipal Institutions (BPHR202203032); R&D Program of Beijing Municipal Education Commission (M202410009014); and Yuxiu Innovation Project of NCUT (Project No. 2024NCUTYXCX107).

Keywords: adaptive continuous deep belief network | health indicator | remaining useful life prediction | SSA-KELM model

ABSTRACT

Rolling bearings are crucial components in a wide variety of machinery. Monitoring their conditions and predicting their remaining useful life (RUL) is vital to prevent unexpected breakdowns, optimize maintenance schedules, and reduce operational costs. This article proposes an approach based on adaptive continuous deep belief networks (ACDBN) and improved kernel extreme learning machine (KELM) to predict the RUL of rolling bearings. In the proposed approach, the ACDBN model is used for extracting hidden fault features and the distance between the initial health state and the real-time degradation state is used to construct a health indicator (HI). Then, a hybrid kernel extreme learning machine prediction model optimized by the sparrow search algorithm (SSA-KELM) is proposed to estimate the RUL using the extracted HIs. The SSA is used to find the optimal parameters of the KELM model. The proposed method has been assessed using existing bearing datasets. The obtained results indicate that the proposed method successfully improves RUL prediction accuracy compared to existing approaches in the literature.

1 | Introduction

Rolling bearings are an essential component of rotating machinery. However, due to their continuous operation under varying loads, speeds, and environmental conditions, rolling bearings are subject to wear and eventual failure, which will affect the operation conditions [1, 2]. Remaining useful life (RUL) prediction of rolling bearings can provide useful information for predictive maintenance allowing to establish an effective plan in advance to improve the operation cycle of all mechanical equipment and prevent fault appearance. Therefore, to enhance the reliability, safety, and efficiency of machinery while reducing maintenance costs and extending the life of critical assets, RUL prediction has attracted a lot of attention from research community.

Existing RUL prediction methods can be divided into model-based, knowledge-based and data-driven methods [3]. Model-based methods consist of using mathematical or physical models to analyze the failure mechanism. The prediction accuracy is heavily dependent on the capability of the mathematical model to characterize the aging process. While in practice, accurate modeling is challenging due to complexities, incomplete data,

Abbreviations: ACDBN, adaptive continuous deep belief networks; HI, health indicator; KELM, kernel extreme learning machine; RUL, remaining useful life.

varying conditions, among others. Besides, using expert knowledge to predict RUL of rolling bears is complicated, especially under complex working conditions, which leads to low accuracy and cannot be updated online.

Alternatively, data-driven methods enable the construction of prediction models based only historical data without requiring physical characterization of the degradation process. Among data-driven methods, statistical analysis and machine learning are two primary classes of algorithms. Statistical analysis methods assume the existence of a degradation model such that the monitoring data can be directly used to calculate model parameters. However, degradation models in practical engineering are difficult to obtain.

In recent decades, machine learning based data-driven methods has become an effective tool for RUL prediction. Based on the amount of data required during the training process, machine learning method can be divided into shallow learning and deep learning. Among the existing shallow learning methods, artificial neural network (ANN) [4] and multilayer perceptron (MLP) [5] have been proposed for RUL prediction. However, they present several issues such as local minima and overfitting. The support vector machine (SVM) can solve the overfitting problem presenting satisfactory generalization capabilities. However, a wrong parameter initialization can negatively affect the prediction results. The extreme learning machine (ELM) is a fast and reliable algorithm that is based on a single-hidden layer forward network. It has good performance with a faster learning speed and stronger generalization than deep learning and other forms of shallow machine learning. For this reason, ELM-based RUL prediction has attracted extensive research [6]. Besides, in recent years, deep learning methods have become increasingly popular for predicting RUL due to their ability to handle complex data and extract meaningful features [7]. In [8], a partial domain adaptation method is proposed for RUL prediction with incomplete target-domain data based on a deep neural network-based adversarial learning strategy.

Fully extracting the feature information reflecting bearing degradation is crucial for the success of data-driven RUL prediction methods. The health index (HI) is a metric used to represent the overall health or condition of a system, component, or piece of equipment. It can effectively captures the progressive degradation of a system, making it a reliable and practical tool for RUL estimation [9]. Furthermore, the constructed HI typically belongs to one-dimensional data, by converting multiple sensor readings and operational data into an HI. In this way, the complexity of the data is reduced, allowing for more straightforward RUL prediction [10]. As a result, the shallow learning method is more suitable for RUL prediction after the HI construction since the amount of data required is smaller. Since kernel extreme learning machine (KELM) has the advantage of fast training, simplicity, good generalization, and robustness against overfitting and local minima, this article focus on the KELM based shallow learning method to predict the RUL of rolling bearings. Note that the prediction accuracy of KELM depends on its kernel parameter and regularization coefficient. How to select them is one of the key issues to design KELM models. Using swarm intelligence optimization algorithm to find the optimal parameters of KELM is considered as one of the effective methods. The sparrow search algorithm (SSA) is a relatively recent optimization algorithm inspired by the foraging behavior and antipredation strategies of sparrows. It is primarily used for solving complex optimization problems in various fields [11]. As a result, in this article, it is used to optimize the kernel parameter and regularization coefficient of KELM.

HI construction using deep learning models can be divided into unsupervised and supervised learning approaches. Generally, supervised methods require time-consuming labeling process, while unsupervised techniques can efficiently extract features from large volumes of unlabeled data. This has led to growing interest in unsupervised methods for HI construction in recent years. Deep belief network (DBN) is an unsupervised learning approach that has been widely used in HI construction. In [12], DBN is combined with self-organizing mapping to extract HI in a strong noise environment. Typical, DBNs are restricted to Boltzmann machines (RBMs), which are particularly well-suited for modeling discrete data, rather than continuous data, due to their random binary units. However, since the rolling bearing data are continuous, the DBN-based HI construction performance may be decreased. To solve this problem, Chen et al. propose a continuous RBM (CRBM) method that can be constructed by including zero-mean Gaussian noise in the visible layer [13]. CRBMs can be used in continuous deep belief network (CDBN) where it is applied for the fault diagnosis of rolling bearings to improve the processing of continuous input data [14]. However, the learning rate in CDBN is fixed. A fixed learning rate makes it difficult to find the most suitable value by experience.

This article proposes a new method for RUL prediction with application to rolling bearings. To solve this challenging problem, three problems have to be addressed: first, to define a HI that can be derived from raw data to assess system health while reducing the dimensionality of the available data. Then, to obtain a RUL prediction model that is able to predict the HI evolution in time. Finally, to tune the model parameters in an optimal manner. The main contributions of this article to address previous problems are summarized as follows:

First, an unsupervised HI construction method is proposed, which is suitable for the situations where labeled data is unavailable or limited. The extracted HI can be used for predicting the rolling bearing's RUL effectively. Then, an adaptive CDBN model is proposed to automatically learn and extract the HIs from raw sensor data of rolling bearings. In this model, an adaptive learning mechanism that can dynamically adjust the learning rate is designed for continuous DBNs to achieve a stable feature extraction process. The proposed mechanism is able to provide faster convergence and better generalization than competing methods in the literature. Furthermore, this strategy can eliminate the need for extensive preprocessing or manual feature selection. Next, based on the extracted HI, a hybrid SSA-KELM model is designed to predict the rolling bearings RUL, in which the key parameters of KELM are optimized by SSA method. The integration of SSA with KELM represents can leverage the strengths of both techniques to enhance prediction accuracy and computational efficiency. Finally, comprehensive experiments with public available rolling bearing data are executed to prove that the proposed ACDBN-SSA-KELM method can achieve

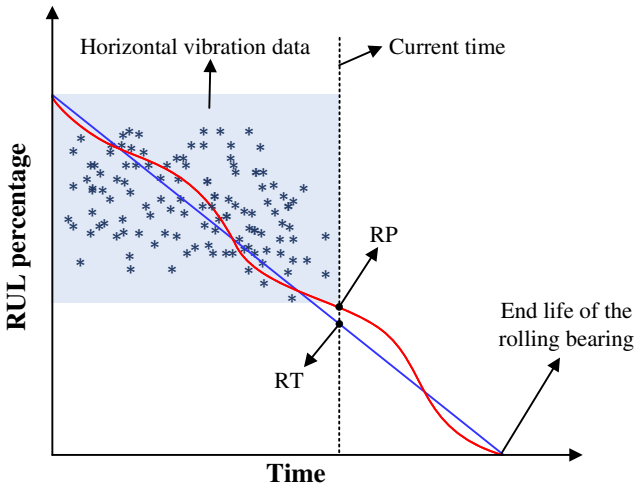

FIGURE 1 | The description of RUL.

higher prediction accuracy, which is required in real industrial applications.

The structure of the remainder of this article is the following: Section 2 introduces the problem statement. In Section 3, the ACDBN-SSA-KELM model and related theories are proposed. In Section 4, experimental results and analysis are presented. In Section 5, conclusions and future works are provided.

2 | Problem Statement

Given a set of horizontal vibration data obtained from a rolling bearing, RUL is defined as the time remaining from the current time to the end of the bearing life cycle and is calculated as [10]:

$$\mathrm{RUL} = T - T_{\mathrm{cur}} \tag{1}$$

where T_{cur} represents age and T represents the full life cycle of the rolling bearing.

According to [15], the predicted and real age percentages at the current moment are defined as

$$\mathrm{RP} = \frac{T_{\mathrm{cur,\,pre}}}{T}, \quad \mathrm{RT} = \frac{T_{\mathrm{cur,\,rea}}}{T} \tag{2}$$

where $T_{\mathrm{cur,\,pre}}$ and $T_{\mathrm{cur,\,rea}}$ represent current moment predicted and real age of the rolling bearing obtained from historical data, respectively. Figure 1 illustrates the temporal evolution of these percentages.

The prediction of the RUL percentage is used as the output of the method proposed in this article. At each time instant, RUL can be predicted from RP as follows

$$\mathrm{RUL}_{\mathrm{pre}} = (1 - \mathrm{RP})T \tag{3}$$

In RUL prediction, the goal is to minimize the difference between RT and RP. This difference is an indicator of the accuracy of the proposed prediction method.

3 | Proposed Approach

This article proposes a method for predicting the RUL of rolling bearings based on an improved ACDBN-SSA-KELM model, which involves leveraging the strengths of the ACDBN feature extraction model and SSA-KELM prediction model. The proposed method for RUL prediction includes two components (see Figure 2):

- *Feature extraction*: A novel adaptive continuous deep belief network (ACDBN) method is proposed to extract depth features from original data and construct HI reflecting bearing degradation.
- *RUL prediction*: The RUL of rolling bearings can be predicted based on HI by using a hybrid SSA-KELM model.

3.1 | A Novel ACDBN Structure

In this section, a novel ACDBN is proposed based on the standard deep belief network method, in which an adaptive mechanism is designed to update the learning rate parameters. The structures of the proposed ACDBN and its corresponding ACRBMs are depicted in Figure 3.

ACRBM includes visible layers with zero mean Gaussian noise and hidden layers. The visible layers are used to receive the input data while the hidden layers are used to extract features. Similar to the standard RBM, the input u_i of each unit in a hidden layer is generated from all unit states in the visible layer. Each hidden unit has a status value s_j that is equal to the weighted sum of all visible units. The relationship between u_i and s_j can be expressed as:

$$s_j = Sig\left(\sum_i w_{ij}u_i + n_j\right) \tag{4}$$

where s_j represents the output of neuron j and u_i represents the input from neurons to other neurons j. w_{ij} is the connection weight of ACRBM. n_j represents the Gaussian noise input component. $Sig(x) = 1/1 + e^{-a_j x}$ is the nonlinear activation function, parameter a_j determines the slope of the activation function.

The probability distribution $p(n_j)$ of Gaussian noise n_j introduced on the basis of DBN is expressed as:

$$n_j = \sigma \cdot N_j(0,1) \tag{5}$$

$$p(n_j) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-n_j^2}{2\sigma^2}\right) \tag{6}$$

where σ is a given constant and $N_j(0,1)$ denotes a Gaussian noise with zero mean and unit variance.

ACRBM is an improved model of RBMs in which an adaptation mechanism is introduced to adjust the noise level. In ACRBM, the determination of connection weight w_{ij} and noise control a_j parameters is accomplished by minimizing contrastive divergence (MCD) algorithm during the training process. MCD serves

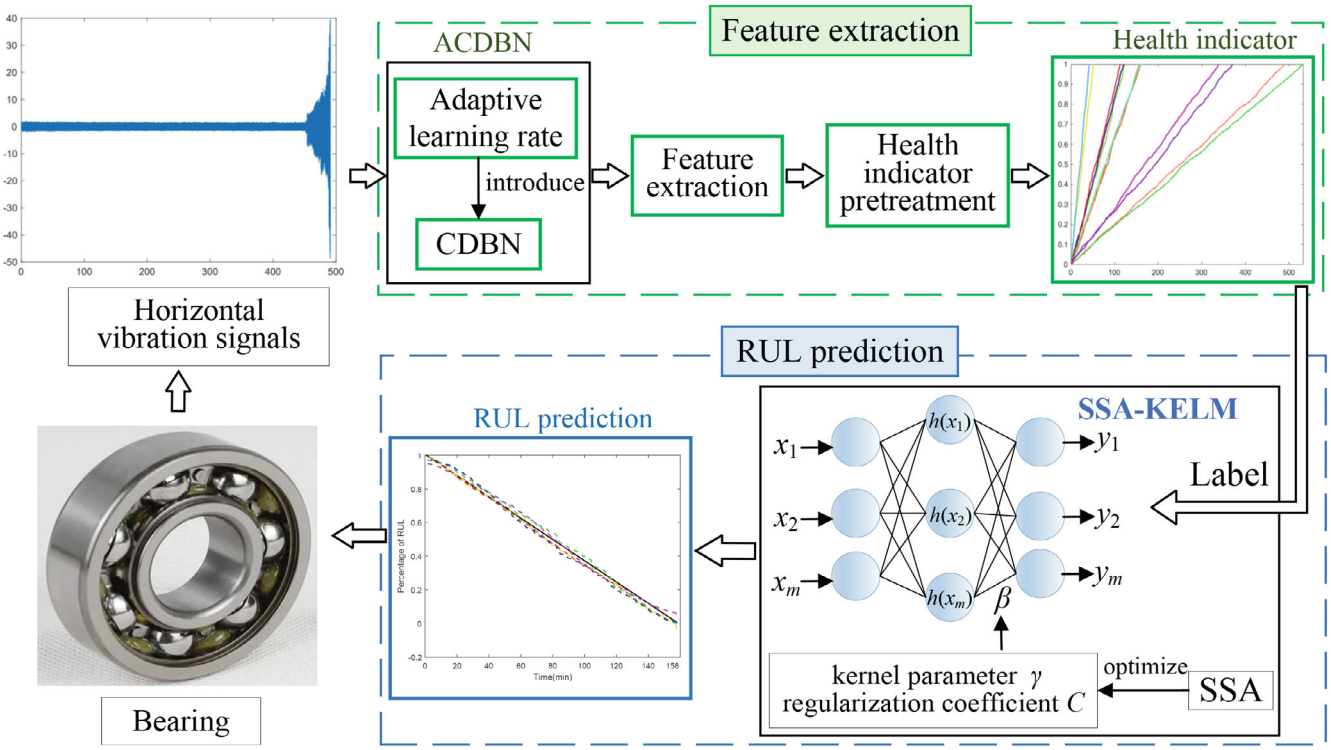

FIGURE 2 | Flow chart of the proposed strategy.

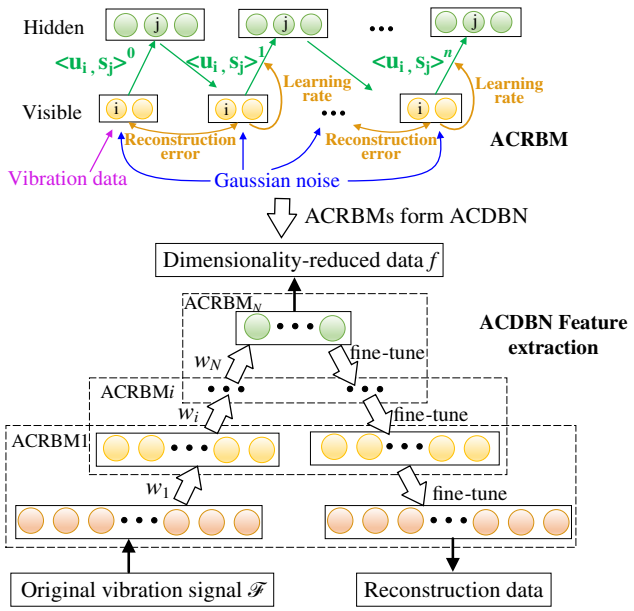

FIGURE 3 | The structure of the proposed ACDBN.

as a training algorithm that iteratively adjusts the connection weight and noise control parameters. This allows the distribution of the model to approximate the distribution of the training data, improving the generative and expressive model capabilities. There are many advantages of MCD specifically in parameter tuning and updating during the training process:

$$w_{ij}^{(\tau+1)} = w_{ij}^{(\tau)} + \Delta w_{ij}^{(\tau+1)} \tag{7}$$

$$a_j^{(\tau+1)} = a_j^{(\tau)} + \Delta a_j^{(\tau+1)} \tag{8}$$

$$\Delta w_{ij}^{(\tau+1)} = \eta_w \left(\left\langle u_i^{(\tau)} s_j^{(\tau)} \right\rangle - \left\langle u_i^{(\tau+1)} s_j^{(\tau+1)} \right\rangle \right) \tag{9}$$

$$\Delta a_j^{(\tau+1)} = \frac{\eta_a}{(a_j^{(\tau)})^2} \left(\left\langle \left(s_j^{(\tau)} \right)^2 \right\rangle - \left\langle \left(s_j^{(\tau+1)} \right)^2 \right\rangle \right) \tag{10}$$

where Δw_{ij} and Δa_j represent the change of two adjacent iterations. η_w and η_a represent the learning rate of connection weights and noise control parameters, respectively. $\langle \cdot \rangle$ represents the data expectation. The training of ACRBM concludes after n iterations.

According to Equations (9) and (10), the learning rate η_w, η_a and reconstructed data $u_i^{(\tau+1)}$, $s_j^{(\tau+1)}$ are necessary for updating the parameters. However, the reconstructed data and reconstruction error change during each iteration. In order to keep the parameters in the optimal state during each iteration, the learning rate must be adjusted automatically. For this reason, this article introduces an adaptive mechanism into the learning rates (η_w, η_a) to reduce the randomness of the model. Reducing the randomness enables the convergence of reconstruction errors.

If the length of the data is set as m, the expression of the reconstruction error is

$$E^{(\tau+1)} = \sum_{i=1}^{m} \left(u_i^{(\tau+1)} - s_i^{(\tau)} \right)^2 \tag{11}$$

where τ denotes the iterations number and $u_i^{(\tau+1)}$ denotes the reconstructed data of the $(\tau+1)$th iteration. The learning rate

parameters are determined by comparing the values of two consecutive reconstruction errors

$$\Delta E^{(\tau+1)} = E^{(\tau+1)} - E^{(\tau)} \tag{12}$$

The adaptive learning rate rule is expressed as:

$$\eta^{(\tau+1)} = \begin{cases} k_a \eta^{(\tau)}, & \Delta E^{(\tau+1)} \geq 0 \\ k_b \eta^{(\tau)}, & \Delta E^{(\tau+1)} < 0 \end{cases} \tag{13}$$

where η represents the learning rate, k_a and k_b represent the attenuation coefficient and growth coefficients, respectively. This means when the reconstruction error becomes larger than that of the previous iteration process, the learning rate is reduced. Generally, the attenuation coefficient k_a is chosen between 0.1 and 0.5. In this way, the learning rate is reduced by 10% to 50% of its current value. Otherwise, when the reconstruction error becomes smaller than that of the previous iteration, the learning rate must be increased by multiplying a growth coefficient k_b. The growth factor is generally set between 1.05 and 2.0, meaning the learning rate increases by 5% to 100%.

The adaptive learning rate rule can not only improve the robustness of the network, but also accelerate the convergence process. According to Equation (13), the learning rates η_w and η_a are automatically adjusted according to the adaptive rules until the end of the iteration.

During ACDBN training, neurons in every two adjacent layers (from the bottom to the top) serve as an ACRBM. The output of the ACRBM of the upper layer serves as the input of the ACRBM of the next layer. It is possible to reduce data dimension for a single ACRBM when there are fewer neurons in the hidden layer than in the visible layer. The minimum covariance determinant (MCD) algorithm is used to determine the connection weight and noise control parameters of each ACRBM, which is a method for estimating the mean and covariance matrix in a way that tries to minimize the influence of anomalies. Then, the algorithm is used to calculate the state of neurons s_j in each layer. The result of the calculation is used to track the degradation state information during the operation of the rolling bearings.

3.2 | HI Construction Based on ACDBN

In this study, the original data is input to the proposed ACDBN network to construct the HI. The objective is to construct HI by extracting the desired features from the original data. Supposing that the input of ACDBN is a matrix of vibration signals $\mathcal{F} \in R^{v \times m}$, which is expressed as

$$\mathcal{F} = \begin{bmatrix} F_{11} & F_{12} & \cdots & F_{1m} \\ F_{21} & F_{22} & \cdots & F_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ F_{v1} & F_{v2} & \cdots & F_{vm} \end{bmatrix} \tag{14}$$

where each column $F_t \in R^v$, $t = 1, \ldots, m$ represents the vibration signal data obtained from one sampling, there are v vibration data obtained from each sampling, and a total of m samples are carried out.

After the structure of the ACDBN model is determined, the number of layers and the nodes of each layer should be determined. Through the model, the v dimension input data can be projected into q dimensional feature subspace with $q < v$, such that a high-dimensional raw vibration data into low-dimensional ones. Then through the ACDBN, the input data is projected to a compressed new feature data $f \in R^{q \times m}$. In this projection process, an appropriate feature layer size should be selected to avoid overfitting. Thus, the constructed HI model is more accurate and robust to noises.

In the next step, the feature data f is used to construct HI, which can measure the degree of equipment deviating from the initial health state in the process of degradation as follows

$$f_{\mathrm{HI},t} = \sqrt{\sum_{t=1}^{m} (f_t - f_1)^2}, \quad t = 1, \ldots, m \tag{15}$$

where f_t denotes the tth output of ACDBN, which is the real-time degradation feature, and f_1 represents the first column of ACDBN output that represents the initial health state.

The generated HI can be rewritten as

$$f_{\mathrm{HI}} = [f_{\mathrm{HI},1}, \ldots, f_{\mathrm{HI},t}, \ldots, f_{\mathrm{HI},m}], \quad t = 1, \ldots, m \tag{16}$$

where $f_{\mathrm{HI},t}$ denotes the HI evaluated at sampling time t.

In order to calculate HI evaluation indicators, the exponential weighted moving average (EWMA) method is used to divide the HI order ($f_{\mathrm{HI},t}$) into two components: the stationary trend term $f^S_{\mathrm{HI},t}$ and the random margin term $f^R_{\mathrm{HI},t}$. According to [16], we have:

$$f^S_{\mathrm{HI},t} = p f_{\mathrm{HI},t} + (1-p) f_{\mathrm{HI},t-1} \tag{17}$$

$$f_{\mathrm{HI},t} = f^S_{\mathrm{HI},t} + f^R_{\mathrm{HI},t} \tag{18}$$

where p $(0 < p < 1)$ represents the weighting coefficient of the EWMA filter.

Time correlation and monotonicity are the commonly used indicators to evaluate the performance of HI [16]. In this article, these two indicators are considered:

Time correlation: The equipment degradation evolves gradually with the increase of running time. Measures of the trend of HI are based on the correlation coefficient between HI and time, which is evaluated as:

$$\mathrm{Corr}(f_{\mathrm{HI},t}, t) = \frac{\left| \sum_{t=1}^{m} \left(f_{\mathrm{HI},t} - \overline{f_{\mathrm{HI}}} \right)(t - \bar{t}) \right|}{\sqrt{\sum_{t=1}^{m} \left(f_{\mathrm{HI},t} - \overline{f_{\mathrm{HI}}} \right)^2 \sum_{t=1}^{m} (t - \bar{t})^2}} \tag{19}$$

where $f_{\mathrm{HI},t}$ represents the HI of the tth points; $\overline{f_{\mathrm{HI}}}$ and $\bar{t}$ are the mean value of $f_{\mathrm{HI},t}$ and t, respectively.

Monotonicity: The monotonicity evaluation index assesses the tendency of the HI. A high score indicates a strong performance in monotonicity. Monotonicity is evaluated as:

$$Mon(f_{\mathrm{HI},t}) = \frac{1}{m-1}\left|\sum_{t=1}^{m}\delta\left(f^{S}_{\mathrm{HI},t+1} - f^{S}_{\mathrm{HI},t}\right) - \sum_{t=1}^{m}\delta\left(f^{S}_{\mathrm{HI},t} - f^{S}_{\mathrm{HI},t+1}\right)\right| \tag{20}$$

where $\delta(\cdot)$ is a unit step function, which is expressed as:

$$\delta(x) = \begin{cases} 1, & x \geqslant 0 \\ 0, & x \leqslant 0 \end{cases} \tag{21}$$

The value range of the two evaluation indexes is [0,1]. This range is positively correlated with the performance of the constructed HI.

3.3 | Hybrid SSA-KELM Model

After the HI is constructed, the RUL of rolling bearings can be predicted using it. In this article, a hybrid SSA-KELM method is designed for the RUL prediction of rolling bearings.

Compared with a traditional neural network, the ELM method has a better nonlinear fitting ability and faster learning speed [17]. In this article, the problem of RUL prediction is addressed by ELM method. In this way, the input is HI, which is obtained through Equation (15) and the output $\mathrm{y} = [\mathrm{y}_1, \ldots, \mathrm{y}_j, \ldots, \mathrm{y}_m]$ is the RUL percentage (3), which is expressed as:

$$\mathrm{y}_j = \sum_{i=1}^{q}\beta_{ij}h_i(\omega_{ji}f_{\mathrm{HI},j} + b_i), \quad j = 1, 2, \ldots, m \tag{22}$$

$$\beta = \begin{bmatrix} \beta_{11} & \cdots & \beta_{1m} \\ \beta_{21} & \cdots & \beta_{2m} \\ \cdots & \cdots & \ldots \\ \beta_{q1} & \cdots & \beta_{qm} \end{bmatrix}_{q\times m} \tag{23}$$

$$H = \begin{bmatrix} h_1(w_{11}f_{\mathrm{HI},1} + b_1) & \cdots & h_q(w_{1q}f_{\mathrm{HI},1} + b_q) \\ h_1(w_{21}f_{\mathrm{HI},2} + b_1) & \cdots & h_q(w_{2q}f_{\mathrm{HI},2} + b_q) \\ \cdots & \cdots & \cdots \\ h_1(w_{m1}f_{\mathrm{HI},m} + b_1) & \cdots & h_q(w_{mq}f_{\mathrm{HI},m} + b_q) \end{bmatrix}_{m\times q} \tag{24}$$

$$\beta = H^{T}\left(HH^{T}\right)^{-1}\mathrm{y} \tag{25}$$

where $h_i(\cdot)$ represents the activation function, q represents the number of hidden layer neurons, H represents the hidden layer output matrix, $\beta \in R^{q\times m}$ represents the output weight matrix, and y represents the true percentage of life.

Since the input weight w_{ji} and hidden neuron threshold b_i are randomly generated in ELM, the effect of the method is unstable. In order to improve the accuracy and stability of the method, the kernel parameter is introduced into the algorithm. KELM is an improved version of the ELM model. The unknown nonlinear feature mapping of the hidden layer is represented by a kernel function. This function can enhance the generalization ability and stability of ELM. The parameter $1/C$ is added to the unit diagonal matrix HH^T so that its characteristic root is not zero. The addition of this parameter improves the stability and generalization of KELM. The model of KELM is as follows:

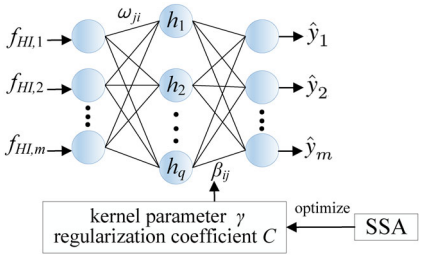

FIGURE 4 | SSA-KELM model structure.

$$\begin{aligned} \hat{\mathrm{y}} &= H\beta \\ &= HH^{T}\left(\frac{1}{C} + HH^{T}\right)^{-1}\mathrm{y} \end{aligned} \tag{26}$$

$$HH^{T} = [K(f_{\mathrm{HI}}, f_{\mathrm{HI},1}), \ldots, K(f_{\mathrm{HI}}, f_{\mathrm{HI},m})] \tag{27}$$

$$K(f_{\mathrm{HI},i}, f_{\mathrm{HI},j}) = \exp\left(-\frac{\left\|f_{\mathrm{HI},i} - f_{\mathrm{HI},j}\right\|^{2}}{\gamma^{2}}\right) \tag{28}$$

where $K(\cdot)$ represents the kernel function (which is usually set as RBF kernel function), C represents the regularization coefficient, and γ represents the kernel parameter.

The selection of the kernel parameter and regularization coefficient in KELM strongly influences the accuracy of the model prediction. That is why the sparrow search algorithm (SSA) will be combined with KELM to optimize the kernel parameter and regularization coefficient. The structure of the SSA-KELM prediction model is displayed in Figure 4.

SSA is a swarm intelligence optimization algorithm that demonstrates to be a very efficient optimization method with fast convergence speed. It mimics the foraging and antipredatory behavior of sparrows. It is particularly useful for solving continuous, nonlinear, and high-dimensional optimization problems. According to the steps of the SSA, the optimization process of the KELM parameters using SSA are given as follows:

First, initialize the number of sparrows P, the kernel parameter γ, and the regularization coefficient C randomly. Then calculate the individual fitness F_P of sparrows $P = 1, \ldots, p$ to determine the current best and worst individuals. The fitness function is defined as:

$$J_P = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{\mathrm{y}}_i - \mathrm{y}_i)^2} \tag{29}$$

where y_i denotes true RUL and $\hat{\mathrm{y}}_i$ denotes the predicted RUL.

ALGORITHM 1 | RUL prediction algorithm based on ACDBN-SSA-KELM approach

Require: Raw vibration data.

for $\tau = 1 : n$ **do**
  Initialize the weight w_{ij} and noise control parameter a_j.
  if $E^{(\tau+1)} - E^{(\tau)} \geq 0$ **then**
    $\eta^{(\tau+1)} = k_a \eta^{(\tau)}$;
  else
    $\eta^{(\tau+1)} = k_b \eta^{(\tau)}$;
  end if
  Calculate Δw_{ij}, Δa_j using Equation (9) and (10).
  Update the weight w_{ij} and noise control parameter a_j.
end for
Calculate the hidden layer output s_j using Equation (4).
The current s_j serves as the u_i of the next ACRBM.
Repeat Steps 1-12 until the last ACRBM is met.
Construct HI using Equation (15).
Add the HI input to the SSA-KELM network.
Initialize the population P, iterations I, and the value range of γ, C.
for $P = 1 : p$ **do**
  Calculate the fitness value according to Equation (29).
end for
Rank p fitness values to obtain the individual optimal fitness.
for $i = 1 : M$ **do**
  Update the finder, follower, and lookout positions using Equation (30)–(32), respectively.
  Repeat Steps 19-22 to obtain the optimal fitness value.
end for
Obtain optimal parameters γ, C when the termination criteria are met.
Insert the optimized parameters γ, C into Equation (26) for training.
Use the trained point prediction model to estimate the test bearing.
Obtain point RUL prediction results.

In the ith cycle, the location of the producer is updated by:

$$x_{P,i} = \begin{cases} x_{P,i} \cdot \exp\left(\frac{-P}{i_{\max}}\right), & R_2 < ST \\ x_{P,i} + Q \cdot D, & R_2 \geq ST \end{cases} \tag{30}$$

where $x_{P,i}$ denotes the Pth sparrow position in the ith cycle. Q represents random number that follows a normal distribution. D represents a $1 \times n$ matrix in which each element value is one. $ST \in [0.5, 1]$ and $R_2 \in [0, 1]$ represent the safety threshold and alarm value, respectively. If $R_2 < ST$, finders can continue their search in the absence of predators. Otherwise, it indicates that sparrows have been found by the predator and that all sparrows need to fly to safety. $i_{\max}$ represents maximum iteration.

Since the consumers adjust their positions by following the producers, then the positions of the consumer are updated by:

$$x_{P,i} = \begin{cases} Q \cdot \exp\left(\frac{x_{\text{bad}} - x_{P,i}}{P^2}\right), & P > N_1/2 \\ x_b + \left|x_{P,i} - x_b\right| \cdot B^+ \cdot D, & \text{otherwise} \end{cases} \tag{31}$$

where x_{bad} reflects the finder's worst global position and x_b reflects the finder's best global position, $B \in R^{1 \times n}$ and $B^+ = B^T \left(BB^T\right)^{-1}$.

Next, sparrows with lower fitness values increase their alertness and may move toward safer areas. This behavior helps maintain diversity in the population and avoid premature convergence to local optima. Then, the positions of the sparrows are updated by:

$$x_{P,i} = \begin{cases} x_{best} + \beta_a \cdot \left|x_{P,i} - x_{best}\right|, & f_P > f_g \\ x_{P,i} + l \cdot \left(\frac{\left|x_{P,i} x_{best}\right|}{(f_P - f_w) + \varepsilon_a}\right), & f_P = f_g \end{cases} \tag{32}$$

where x_{best} represents the most optimal position and β_a represents the step control parameter. $l \in [-1, 1]$ is a random number. ε_a is a constant to prevent a denominator of zero. f_P is the fitness value of the current sparrow. f_w and f_g is the worst and best global fitness values, respectively. $f_P > f_g$ means that sparrows are at the edge of the group. The sparrows at the edge of the group are more likely to encounter predators. $f_P = f_g$ illustrates that sparrows are at the center of a population that needs to approach other sparrows for antipredation.

The SSA algorithm iterates through the steps of updating producer and consumer positions and incorporating awareness of danger until a stopping criterion is met (e.g., a maximum number of iterations or convergence to a satisfactory solution). Finally, the RUL prediction based on the proposed ACDBN-SSA-KELM approach is summarized in Algorithm 1.

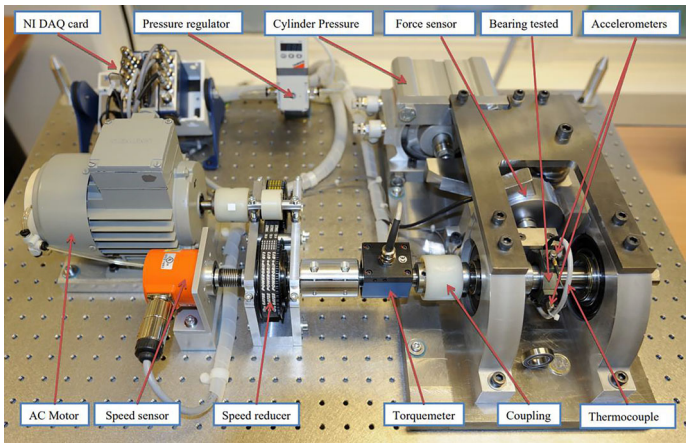

FIGURE 5 | Test platform of PRONOSTIA.

TABLE 1 | Experimental data (PHM2012 bearing datasets).

Working condition	Condition 1	Condition 2	Condition 3
Rotating speed (rpm)	1800	1650	1500
Loading force (N)	4000	4200	5000
Training dataset (full life cycle)	Bearing 1–1 (28,030 s)	Bearing 2–1 (9110 s)	Bearing 3–1 (5150 s)
	Bearing 1–2 (8710 s)	Bearing 2–2 (7970 s)	Bearing 3–2 (16,370 s)
Testing dataset (full life cycle)	Bearing 1–3 (23,750 s)	Bearing 2–3 (19,550 s)	Bearing 3–3 (4340 s)
	Bearing 1–4 (14,280 s)	Bearing 2–4 (7510 s)	
	Bearing 1–5 (24,630 s)	Bearing 2–5 (23,110 s)	
	Bearing 1–6 (24,480 s)	Bearing 2–6 (7010s)	
	Bearing 1–7 (22,590 s)	Bearing 2–7 (2300s)	

4 | Results

This section demonstrates the effectiveness of the proposed RUL prediction strategy on the PHM2012 bearing dataset [18] and the XJTU-SY bearing dataset [15].

4.1 | Data Description

4.1.1 | PHM2012 Bearing Dataset

The PHM2012 data was collected by two acceleration sensors in vertical and horizontal directions. The experimental system named PRONOSTIA is shown in Figure 5. Data were recorded every 10 s. There was a recording time of 0.1 s and a sampling frequency of 25.6 kHz. Horizontal vibration data are used in this article.

As shown in Table 1, the whole life data of bearings 1–1, 1–2, 2–1, 2–2, 3–1, and 3–2 are the training set in this article. The whole life data of the remaining 11 bearings (1–3, 1–4, 1–5, 1–6, 1–7, 2–3, 2–4, 2–5, 2–6, 2–7, and 3–3) are used as a test set for RUL prediction experiments.

4.1.2 | XJTU-SY Bearing Dataset

The XJTU-SY dataset collection was carried out by two acceleration sensors, which are located in vertical and horizontal directions. The experimental test platform is shown in Figure 6. Data points were recorded every 1 min. Each recording time is 1.28 s. The sampling frequency is 25.6 kHz. The horizontal vibration data are used for the experiment.

As displayed in Table 2 (following a similar division form of the PHM2012 bearing dataset), the whole life data of bearing 1–1, 1–2, 2–1, 2–2, 3–1, and 3–2 are used as the training set. The whole life data of the remaining nine bearings (1–3, 1–4, 1–5, 2–3, 2–4, 2–5, 3–3, 3–4, and 3–5) are used as the test set.

4.2 | Evaluation Metrics

Root mean square error (RMSE) and the error between the actual value and the predicted value (Er) are used to evaluate the RUL prediction. For all test data, RMSE can measure the disparity between the predicted result $\hat{y}_i$ and the true value y_i.

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2} \tag{33}$$

The bearing failure reaches the end of life when the life percentage reaches 1. The predicted RUL of the ith bearing is obtained by subtracting the current life RP from the total life cycle as RQ

$$\mathrm{RUL}_i = \mathrm{RQ}_i - \mathrm{RP}_i \tag{34}$$

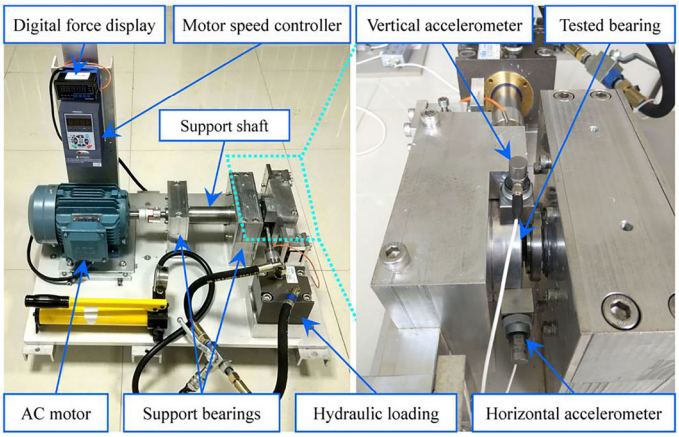

FIGURE 6 | Test platform of XJTU-SY.

TABLE 2 | Experimental data (XJTU-SY bearing datasets).

Working condition	Condition 1	Condition 2	Condition 3
Rotating speed (rpm)	2100	2250	2400
Loading force (kN)	12	11	10
Training dataset (full life cycle)	Bearing 1–1 (123 min)	Bearing 2–1 (491 min)	Bearing 3–1 (2538 min)
	Bearing 1–2 (161 min)	Bearing 2–2 (161 min)	Bearing 3–2 (2496 min)
Testing dataset (full life cycle)	Bearing 1–3 (158 min)	Bearing 2–3 (533 min)	Bearing 3–3 (371 min)
	Bearing 1–4 (122 min)	Bearing 2–4 (42 min)	Bearing 3–4 (1515 min)
	Bearing 1–5 (52 min)	Bearing 2–5 (339 min)	Bearing 3–5 (114 min)

TABLE 3 | ACDBN and SSA-KELM network parameters.

ACDBN	Parameters	SSA-KELM	Parameters
ACRBM layer	4	Kernel parameter upper bound	10
Network structure	41-28-14-5	Kernel parameter lower bound	1
Initial weight learning rate	0.005	Kernel type	RBF
Initial noise term learning rate	0.001	Regularization coefficient upper bound	20
Attenuation coefficient k_a	0.1	Regularization coefficient lower bound	1
Growth coefficient k_b	2	Population number	20
Training number	50	Iterations	50

The prediction performance of the RUL is represented by the error Er_i between RUL_i and ActRUL_i as

$$\mathrm{Er}_i = \frac{\mathrm{ActRUL}_i - \mathrm{RUL}_i}{\mathrm{ActRUL}_i} \times 100\% \tag{35}$$

4.3 | Experimental Process and Results Analysis

To extract degradation features, the bearing data training set is normalized and substituted into the ACDBN network. First, HI is calculated using Equation (15) and then integrated into SSA-KELM for RUL prediction. The parameter settings of ACDBN and SSA-KELM are displayed in Table 3. The process of the experiment is as follows:

Data preprocessing: Since ACDBN requires input data in the range of $[0, 1]$, the min–max normalization method must be used to process the data and scale the data between 0 and 1. Given the original signal $\mathscr{F}$ (14), the normalized value can be obtained

$$\mathscr{F}_t' = \frac{\mathscr{F}_t - \min(\mathscr{F})}{\max(\mathscr{F}) - \min(\mathscr{F})} \tag{36}$$

where $\mathscr{F}_t$ represents the data at sampling time t, $\min(\mathscr{F})$ and $\max(\mathscr{F})$ represent the minimum and maximum values of data at all times, respectively.

HI is extracted by ACDBN: The ACDBN comprises four ACRBMs to extract features in the simulations. The input multidimensional data is mapped to 1-dimensional deep degradation features to achieve data dimension reduction. When the

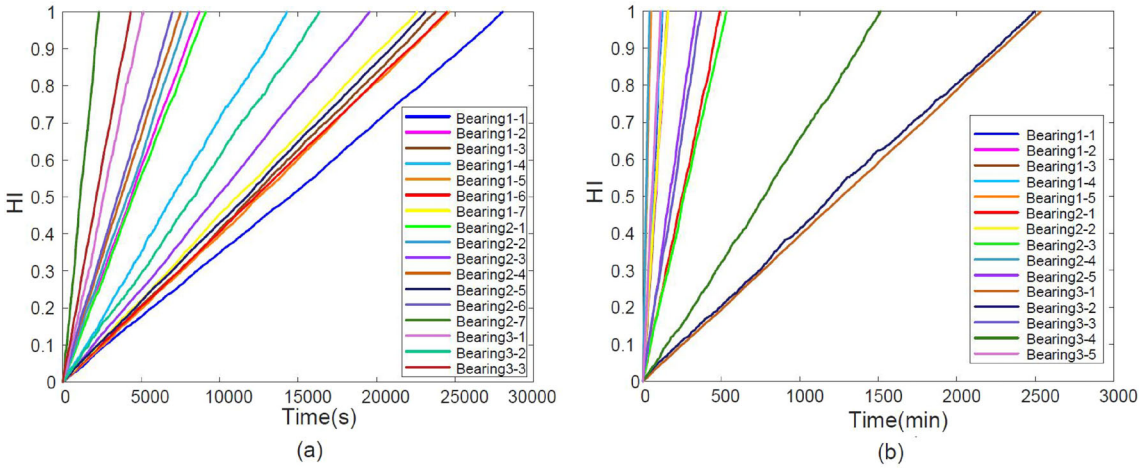

FIGURE 7 | HI of (a) PHM2012 dataset, (b) XJTU-SY dataset.

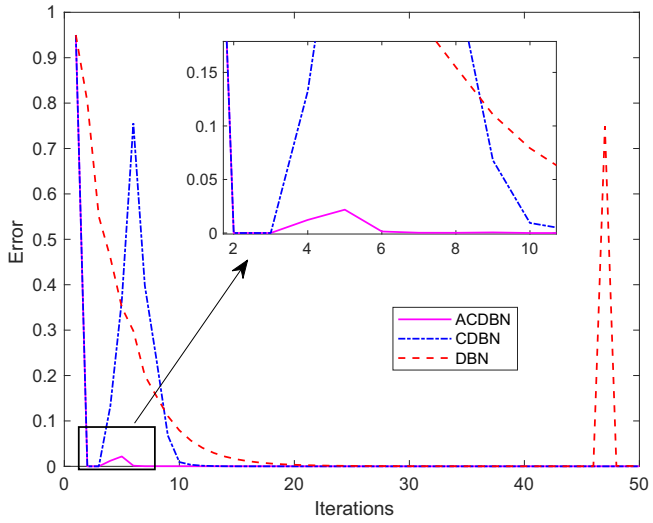

FIGURE 8 | Error comparison results.

parameters are set (as shown in Table 3), the reconstruction error is minimum. HI of each bearing in XJTU-SY bearing dataset ang PHM2012 bearing dataset are shown in Figure 7.

In order to demonstrate that ACDBN has an appropriate convergence behavior and the feature extraction process is stable after the introduction of the adaptive learning rate, errors of the first layer of ACRBM, CRBM, and RBM are compared in the training process. The results of this comparison are presented in Figure 8. In Figure 8, ACDBN quickly converges and remains stable. Significant fluctuation were observed in the case of the other two methods. Note that the competing methods do not present a good convergence behavior.

The proposed ACDBN algorithm is compared with some similar existing ones in the literature: CDBN [14], DBN [10], and SAE [19]. The indicators (time correlation, monotonicity, and robustness average results) for test sets are presented in Table 4. This table demonstrates that the evaluation index of HI extracted by ACDBN is superior to other methods. Consequently, it provides a better estimate equipment degradation from monitoring data.

RUL prediction for PHM2012 bearing dataset: In the training stage, bearings 1–1, 1–2, 2–1, 2–2, 3–1, and 3–2 are the features extracted by ACDBN to construct HI. They are then used as inputs of the SSA-KELM model. The SSA-KELM prediction model is trained using the real-life percentage as the output of the model. The kernel parameter and regularization coefficient are optimized by SSA. The parameter setting in the model is displayed in Table 3.

In the testing stage, other bearings data sets are taken into consideration. They are similar but not the same used in the training stage. After ACDBN feature extraction and HI construction, bearing data are used as inputs of the trained SSA-KELM prediction model to predict life percentage. Figure 9 presents the RUL of all testing bearings predicted by the proposed method and other methods used for comparison.

Based on the PHM2012 bearing dataset, the prediction errors and average scores of other prediction methods are displayed in Table 5. To further demonstrate the effectiveness of the proposed method, the results have also be compared with some existing results. Table 6 shows the comparison result between the proposed method and some other methods published in existing references. The results show that the proposed method has less error and higher scores than others. Additionally, the average prediction error of the proposed method is lower than the average prediction error presented in the work of [22–25]. Hence, the efficiency of the proposed method is further assessed.

Besides, the performance of the proposed method can be demonstrated through the scoring formula of the PHM2012 bearing dataset. The average scores, presented in Table 7, are calculated as follows

$$\text{Score} = \frac{1}{11}\sum_{i=1}^{11}(A_i) \tag{37}$$

where

$$A_i = \begin{cases} \exp^{-\ln(0.5)\cdot(\text{Er}_i/5)}, (\text{Er}_i \le 0) \\ \exp^{+\ln(0.5)\cdot(\text{Er}_i/20)}, (\text{Er}_i > 0) \end{cases} \tag{38}$$

TABLE 4 | HI evaluation comparison results.

HI construction method	ACDBN	CDBN	DBN	SAE	SE-ConvLSTM [20]	DCRM [21]
Correlation	0.998	0.996	0.992	0.923	0.981	0.953
Monotonicity	0.996	0.985	0.952	0.792	0.976	0.266

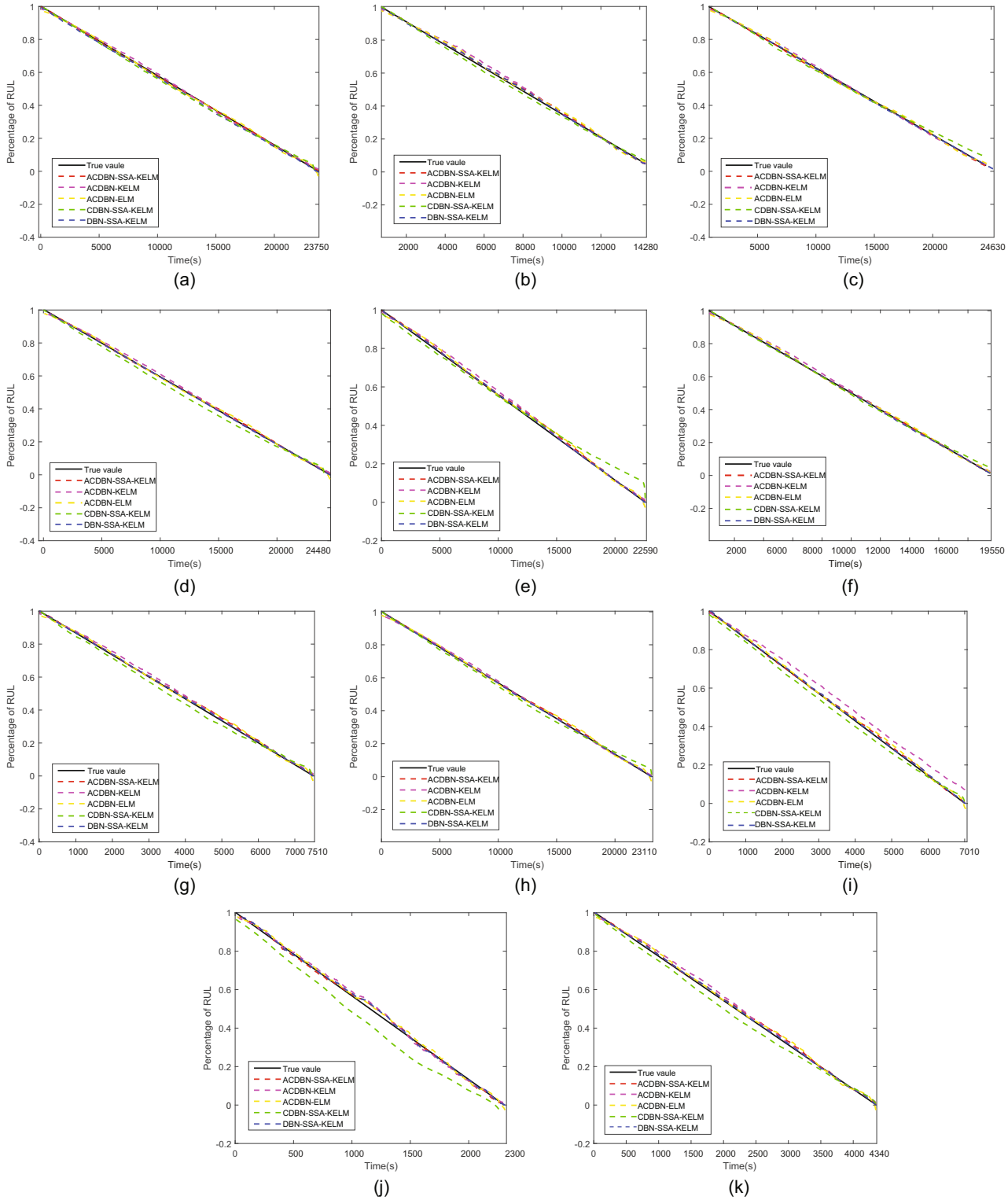

FIGURE 9 | RUL prediction results of different methods in the PHM2012 dataset. (a) Bearing 1–3 in the PHM2012 dataset, (b) Bearing 1–4 in the PHM2012 dataset, (c) Bearing 1–5 in the PHM2012 dataset, (d) Bearing 1–6 in the PHM2012 dataset, (e) Bearing 1–7 in the PHM2012 dataset, (f) Bearing 2–3 in the PHM2012 dataset, (g) Bearing 2–4 in the PHM2012 dataset, (h) Bearing 2–5 in the PHM2012 dataset, (i) Bearing 2–6 in the PHM2012 dataset, (j) Bearing 2–7 in the PHM2012 dataset, (k) Bearing 3–3 in the PHM2012 dataset.

TABLE 5 | RMSE of the RUL prediction results on the PHM2012 bearing dataset.

Bearing	ACDBN-SSA-KELM	ACDBN-KELM	ACDBN-ELM	CDBN-SSA-KELM	DBN-SSA-KELM
1–3	0.0028	0.0054	0.0076	0.0053	0.0564
1–4	0.0064	0.0110	0.0131	0.0158	0.0189
1–5	0.0039	0.0044	0.0097	0.0123	0.0585
1–6	0.0032	0.0060	0.0090	0.0071	0.0069
1–7	0.0087	0.0105	0.0122	0.0496	0.0435
2–3	0.0028	0.0057	0.0088	0.0024	0.0234
2–4	0.0095	0.0106	0.0121	0.0123	0.0138
2–5	0.0055	0.0075	0.0098	0.0141	0.0226
2–6	0.0065	0.0086	0.0106	0.0763	0.020
2–7	0.0072	0.0083	0.0099	0.0892	0.0672
3–3	0.0068	0.0078	0.0108	0.0715	0.0671

TABLE 6 | RMSE of RUL predictions for PHM2012 bearing dataset using other methods in the literature.

Bearing	ACDBN-SSA-KELM	CDBN-LSTM	SAGCN-SA [26]	CLSTMF [27]	LSTM [28]	LSTM+ Attention+HF [29]
1–3	0.0028	0.066	0.0420	0.1685	0.2673	0.2802
1–4	0.0064	0.021	0.1210	0.1581	0.2683	0.3101
1–5	0.0039	0.254	0.0750	0.1369	0.2504	0.2616
1–6	0.0032	0.064	0.0590	0.1484	0.2673	0.3230
1–7	0.0087	0.178	0.0380	0.1878	0.2681	0.2234
2–3	0.0028	0.142	0.0690	0.9970	0.3239	0.3399
2–4	0.0095	0.197	0.2440	1.6367	0.3433	0.3486
2–5	0.0055	0.196	0.2290	0.7317	0.3512	0.3626
2–6	0.0065	0.024	0.0990	0.6805	0.3154	0.3297
2–7	0.0072	0.263	0.2550	0.7638	0.3273	0.3315
3–3	0.0068	0.287	0.1060	0.7788	0.2274	0.2474

TABLE 7 | Performance comparison results of the PHM2012 bearing dataset.

Bearing dataset	Current moment (s)	Real RUL (s)	Predicted RUL (s)	Er (%)					
				ACDBN-SSA-KELM	GA+ SVR [30]	RNN-HI [24]	RVM+ Fréchet [22]	RMLE [23]	SOM [25]
1–3	18,010	5730	5910	−1.04	−11.13	43.28	−1.05	2.58	−1.04
1–4	11,380	2890	2830	2.07	−58,41	67.55	−17.99	−9.14	−20.94
1–5	23,010	1610	1500	0.99	−23.66	−22.98	21.74	−0.99	−278.26
1–6	23,010	1460	1440	1.37	−45.55	21.23	6.16	6.03	19.18
1–7	15,010	7570	7510	0.79	17.85	17.83	7.79	−0.70	−7.13
2–3	12,010	7530	1580	−0.66	21.91	37.84	43.03	55.44	10.49
2–4	6110	1390	1350	1.42	10.36	−19.42	1.44	15.56	51.80
2–5	20,010	3090	3060	1.97	−9.77	54.37	18.77	49.19	28.8
2–6	5710	1290	1270	2.25	59.46	−13.95	2.33	38.53	−20.93
2–7	1710	580	600	−3.45	−13.97	−55.17	−3.45	5.17	44.83
3–3	3510	820	770	2.54	41.59	3.66	13.41	2.56	−3.66
Average error				**1.69**	28.51	32.48	12.47	16.90	44.28
Source				**0.9044**	0.2477	0.2631	0.6234	0.6101	0.3551

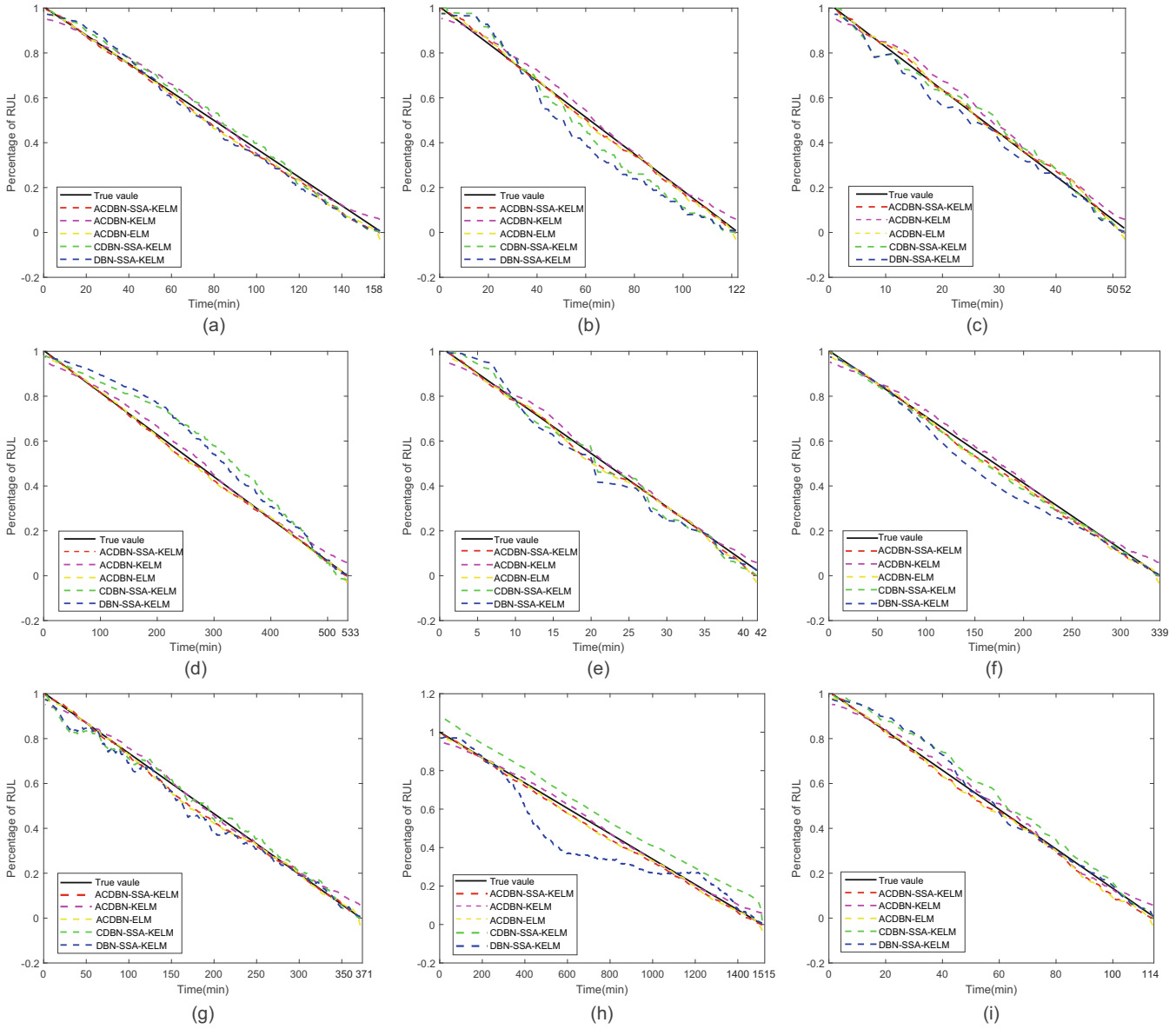

FIGURE 10 | RUL prediction results of different methods in the XJTU-SY dataset. (a) Bearing 1–3 in the XJTU-SY dataset, (b) Bearing 1–4 in the XJTU-SY dataset, (c) Bearing 1–5 in the XJTU-SY dataset, (d) Bearing 2–3 in the XJTU-SY dataset, (e) Bearing 2–4 in the XJTU-SY dataset, (f) earing 2–5 in the XJTU-SY dataset, (g) Bearing 3–3 in the XJTU-SY dataset, (h) Bearing 3–4 in the XJTU-SY dataset, (i) Bearing 3–5 in the XJTU-SY dataset.

TABLE 8 | RMSE of the RUL prediction results on the XJTU-SY bearing dataset.

Bearing	ACDBN-SSA-KELM	ACDBN-KELM	ACDBN-ELM	CDBN-SSA-KELM	DBN-SSA-KELM
1–3	0.024	0.025	0.025	0.027	0.031
1–4	0.003	0.025	0.027	0.325	0.584
1–5	0.015	0.032	0.019	0.120	0.207
2–3	0.006	0.021	0.034	0.369	0.386
2–4	0.016	0.017	0.021	0.259	0.355
2–5	0.011	0.017	0.021	0.066	0.253
3–3	0.020	0.021	0.023	0.101	0.199
3–4	0.006	0.019	0.020	0.123	0.682
3–5	0.019	0.021	0.021	0.220	0.238

TABLE 9 | RMSE of RUL predictions for XJTU-SY bearing dataset using other methods in the literature.

Bearing	ACDBN-SSA-KELM	CDBN-LSTM	CLSTMF [27]	LSTM [28]	LSTM+Attention+HF [29]
1–3	0.024	0.098	0.181	0.237	0.206
1–4	0.003	0.223	0.181	0.259	0.202
1–5	0.015	0.156	0.068	0.188	0.099
2–3	0.006	0.105	0.121	0.125	0.136
2–4	0.016	0.265	0.061	0.106	0.079
2–5	0.011	0.133	0.102	0.137	0.103
3–3	0.020	0.282	0.105	0.122	0.138
3–4	0.006	0.374	0.050	0.154	0.075
3–5	0.019	0.293	0.144	0.198	0.180

TABLE 10 | Performance comparison of the XJTU-SY bearing dataset.

Bearing dataset	Current moment (min)	Real RUL (min)	Predicted RUL (min)	Er (%)		
				ACDBN-SSA-KELM	DBN-SSA-KELM	ACDBN-ELM
1–3	126	32	33	−3.13	15.63	9.38
1–4	98	24	23	4.17	25	12.5
1–5	42	10	9	10	10	20
2–3	426	107	106	0.93	28.04	10.28
2–4	34	8	9	−12.5	12.5	12.5
2–5	271	68	67	1.47	−5.88	11.76
3–3	297	74	77	−4.05	−17.57	6.76
3–4	1212	303	327	−7.92	−12.87	4.29
3–5	91	24	23	4.35	13.04	8.69

Considering the comparison with other methods, the experimental results of the ACDBN-SSA-KELM prediction methods proposed in this article can be summarized as follows:

- Under the same SSA-KELM prediction model, the prediction error of ACDBN feature extraction is lower than that of the DBN feature extraction model. The lower prediction error indicates that the ACDBN feature extraction model is superior.
- ACDBN feature extraction with the SSA-KELM model displays a lower prediction error than with the KELM model and the ELM model. The smaller prediction error indicates the SSA-KELM prediction model superiority.
- Prediction methods improve prediction accuracy at the expense of computational efficiency. The computational efficiency of the proposed method is similar to other methods. Despite this efficiency, the accuracy is higher than other methods.

RUL prediction for XJTU-SY bearing dataset: To verify the generalization ability of the proposed ACDBN-SSA-KELM model, the XJTU-SY bearing dataset is also used and analyzed. Following the same experimental method of the PHM2012 bearing dataset, ACDBN deep feature extraction, SSA-KELM model construction, and RUL prediction are carried out. Experimental results are depicted in Figure 10 and Tables 8–10. These results lead to the same conclusion with the PHM2012 bearing dataset.

5 | Conclusion

This article proposed a novel ACDBN-SSA-KELM model for the RUL prediction of rolling bearings. First, an ACDBN is implemented to extract the degradation features by converting high-dimensional monitoring data into a lower-dimensional representation. The HI extracted with this method presented better monotonicity, robustness, and correlation results than other more commonly used dimensionality reduction methods. Then, an SSA-KELM model is designed to predict the RUL based on the generated HI. It offers a powerful tool for enhancing the reliability and efficiency of industrial maintenance practices. Finally, experimental results on two rolling bearing datasets indicate that the proposed method can effectively improve the RUL prediction accuracy compared with other methods.

Although satisfactory results are obtained by using the proposed method, the prediction is performed at the known end point of life, which indicates that the amount of data is sufficient. For a data-driven prediction method, more run-to-failure data usually improves the model performance. Future research will focus on developing effective data-driven prediction models in scenarios where monitoring data is insufficient.

Acknowledgments

This article is funded by National Natural Science Foundation of China (62273007, 62373005); the Project of Cultivation for Young Top-notch Talents of Beijing Municipal Institutions (BPHR202203032); R&D Program of Beijing Municipal Education Commission (M202410009014); and Yuxiu Innovation Project of NCUT (Project No. 2024NCUTYXCX107).

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

1. X. Chen, X. Li, S. Yu, Y. Lei, N. Li, and B. Yang, "Dynamic Vision Enabled Contactless Cross-Domain Machine Fault Diagnosis With Neuromorphic Computing," *IEEE/CAA Journal of Automatica Sinica* 11 (2024): 788–790.

2. D. Zhu, Y. Liu, and Q. Zhu, "Ensemble Average of Enhanced Generalized Envelope Spectrum for Fault Detection of Rolling Element Bearings," *International Journal of Adaptive Control and Signal Processing* 38 (2024): 1386–1402.

3. N. Li, N. Gebraeel, Y. Lei, L. Bian, and X. Si, "Remaining Useful Life Prediction of Machinery Under Time-Varying Operating Conditions Based on a Two-Factor State-Space Model," *Reliability Engineering and System Safety* 186 (2019): 88–100.

4. M. A. Farsi and H. S. Masood, "Statistical Distributions Comparison for Remaining Useful Life Prediction of Components via ANN," *International Journal of Systems Assurance Engineering and Management* 10, no. 3 (2019): 429–436.

5. Z. Kang, C. Catal, and B. Tekinerdogan, "Remaining Useful Life (RUL) Prediction of Equipment in Production Lines Using Artificial Neural Networks," *Sensors* 21, no. 3 (2021): 932.

6. G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme Learning Machine: Theory and Applications," *Neurocomputing* 70, no. 1-3 (2006): 489–501.

7. W. Zhang, H. Hao, and Y. Zhang, "State of Charge Prediction of Lithium-Ion Batteries for Electric Aircraft With Swin Transformer," *IEEE/CAA Journal of Automatica Sinica* (2024): 1–3, https://doi.org/10.1109/JAS.2023.124020.

8. X. Li, W. Zhang, X. Li, and H. Hao, "Partial Domain Adaptation in Remaining Useful Life Prediction With Incomplete Target Data," *IEEE/ASME Transactions on Mechatronics* 29 (2024): 1903–1913.

9. C. Song, K. Liu, and X. Zhang, "Integration of Data-Level Fusion Model and Kernel Methods for Degradation Modeling and Prognostic Analysis," *IEEE Transactions on Reliability* 67, no. 2 (2017): 640–650.

10. K. Peng, R. Jiao, J. Dong, and Y. Pi, "A Deep Belief Network Based Health Indicator Construction and Remaining Useful Life Prediction Using Improved Particle Filter," *Neurocomputing* 361 (2019): 19–28.

11. J. Xue and B. Shen, "A Novel Swarm Intelligence Optimization Approach: Sparrow Search Algorithm," *Systems Science and Control Engineering* 8, no. 1 (2020): 22–34.

12. Y. Pan, R. Hong, J. Chen, and W. Wu, "A Hybrid DBN-SOM-PF-Based Prognostic Approach of Remaining Useful Life for Wind Turbine Gearbox," *Renewable Energy* 152 (2020): 138–154.

13. H. Chen and A. F. Murray, "Continuous Restricted Boltzmann Machine With an Implementable Training Algorithm," *IEEE Proceedings-Vision, Image and Signal Processing* 150, no. 3 (2003): 153–158.

14. H. Shao, H. Jiang, X. Li, and T. Liang, "Rolling Bearing Fault Detection Using Continuous Deep Belief Network With Locally Linear Embedding," *Computers in Industry* 96 (2018): 27–39.

15. Y. Lei, T. Han, B. Wang, N. Li, T. Yan, and J. Yang, "XJTU-SY Rolling Element Bearing Accelerated Life Test Datasets: A Tutorial," *Journal of Mechanical Engineering* 2019, no. 55 (2019): 1–6.

16. B. Zhang, L. Zhang, and J. Xu, "Degradation Feature Selection for Remaining Useful Life Prediction of Rolling Element Bearings," *Quality and Reliability Engineering International* 32, no. 2 (2016): 547–554.

17. I. Salehi, G. Rotithor, G. Yao, and A. P. Dani, "Dynamical System Learning Using Extreme Learning Machines With Safety and Stability Guarantees," *International Journal of Adaptive Control and Signal Processing* 35 (2021): 894–914.

18. P. Nectoux, R. Gouriveau, K. Medjaher, et al., "PRONOSTIA: An Experimental Platform for Bearings Accelerated Degradation Tests," *IEEE International Conference on Prognostics and Health Management* (Colorado: Denver, 2012): 1–8.

19. T. Han, J. Pang, and A. C. Tan, "Remaining Useful Life Prediction of Bearing Based on Stacked Autoencoder and Recurrent Neural Network," *Journal of Manufacturing Systems* 61 (2021): 576–591.

20. S. Yang, Y. Liu, Y. Liao, and K. Su, "A New Method of Bearing Remaining Useful Life Based on Life Evolution and SE-ConvLSTM Neural Network," *Machines* 10, no. 8 (2022): 639.

21. J. Zhou, Y. Qin, J. Luo, and T. Zhu, "Remaining Useful Life Prediction by Distribution Contact Ratio Health Indicator and Consolidated Memory GRU," *IEEE Transactions on Industrial Informatics* 19, no. 7 (2022): 8472–8483.

22. B. Wang, Y. Lei, N. Li, and N. Li, "A Hybrid Prognostics Approach for Estimating Remaining Useful Life of Rolling Element Bearings," *IEEE Transactions on Reliability* 69, no. 1 (2018): 401–412.

23. X. Liu, P. Song, C. Yang, C. Hao, and W. Peng, "Prognostics and Health Management of Bearings Based on Logarithmic Linear Recursive Least-Squares and Recursive Maximum Likelihood Estimation," *IEEE Transactions on Industrial Electronics* 65, no. 2 (2017): 1549–1558.

24. L. Guo, N. Li, F. Jia, Y. Lei, and J. Lin, "A Recurrent Neural Network Based Health Indicator for Remaining Useful Life Prediction of Bearings," *Neurocomputing* 240 (2017): 98–109.

25. S. Hong, Z. Zhou, E. Zio, and K. Hong, "Condition Assessment for the Performance Degradation of Bearing Based on a Combinatorial Feature Extraction Method," *Digital Signal Processing* 27 (2014): 159–166.

26. Y. Wei, D. Wu, and J. Terpenny, "Bearing Remaining Useful Life Prediction Using Self-Adaptive Graph Convolutional Networks With Self-Attention Mechanism," *Mechanical Systems and Signal Processing* 188 (2023): 110010.

27. S. Wan, X. Li, Y. Zhang, S. Liu, J. Hong, and D. Wang, "Bearing Remaining Useful Life Prediction With Convolutional Long Short-Term Memory Fusion Networks," *Reliability Engineering and System Safety* 224 (2022): 108528.

28. M. Ma and Z. Mao, "Deep-Convolution-Based LSTM Network for Remaining Useful Life Prediction," *IEEE Transactions on Industrial Informatics* 17, no. 3 (2020): 1658–1667.

29. Z. Chen, M. Wu, R. Zhao, F. Guretno, R. Yan, and X. Li, "Machine Remaining Useful Life Prediction via an Attention-Based Deep Learning Approach," *IEEE Transactions on Industrial Electronics* 68, no. 3 (2020): 2521–2531.

30. Q. Li, C. Yan, G. Chen, H. Wang, H. Li, and L. Wu, "Remaining Useful Life Prediction of Rolling Bearings Based on Risk Assessment and Degradation State Coefficient," *ISA Transactions* 129 (2022): 413–428.