Estimation of Bearing Remaining Useful Life Based on Multiscale Convolutional Neural Network

Jun Zhu, Nan Chen, *Member, IEEE*, and Weiwen Peng

***Abstract*—Bearing remaining useful life (RUL) prediction plays a crucial role in guaranteeing safe operation of machinery and reducing maintenance loss. In this paper, we present a new deep feature learning method for RUL estimation approach through time frequency representation (TFR) and multiscale convolutional neural network (MSCNN). TFR can reveal nonstationary property of a bearing degradation signal effectively. After acquiring time-series degradation signals, we get TFRs, which contain plenty of useful information using wavelet transform. Owing to high dimensionality, the size of these TFRs is reduced by bilinear interpolation, which are further regarded as inputs for deep learning models. Here, we introduce an MSCNN model structure, which keeps the global and local information synchronously compared to a traditional convolutional neural network (CNN). The salient features, which contribute for RUL estimation, can be learned automatically by MSCNN. The effectiveness of the presented method is validated by the experiment data. Compared to traditional data-driven and different CNN-based feature extraction methods, the proposed method shows enhanced performance in the prediction accuracy.**

***Index Terms*—Bearing, multiscale convolutional neural network (MSCNN), remaining useful life estimation, time frequency representation (TFR).**

Nomenclature

RUL	Remaining useful life.
TFR	Time frequency representation.
MSCNN	Multiscale convolutional neural network.
WT	Wavelet transform.
HI	Health indicator.
SOM	Self-organizing map.
RNN	Recurrent neural network.
SVR	Support vector regression.
SGD	Stochastic gradient descent.
BP	Backward propagation.
MSE	Mean square error.
MAE	Mean absolute error.
NRMSE	Normalized root mean square error.
$x(t)$	One-dimensional input signal.
$\psi(t)$	Mother wavelet function.
$U(\alpha, \beta)$	Continuous wavelet transform.
$L(\Theta)$	Overall MSE.

Manuscript received October 23, 2017; revised January 15, 2018, February 23, 2018, and April 9, 2018; accepted May 3, 2018. Date of publication June 13, 2018; date of current version November 30, 2018. This work was supported by the National Research Foundation, Sembcorp Industries Ltd. and National University of Singapore under the Sembcorp-NUS Corporate Laboratory with Grant R-261-513-003-281. *(Corresponding author: Weiwen Peng.)*

The authors are with the Department of Industrial Systems Engineering and Management, Sembcorp-NUS Corporate Laboratory, Faculty of Engineering, National University of Singapore, Singapore 119077 (e-mail: isezj@nus.edu.sg; isecn@nus.edu.sg; wwpeng@nus.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2018.2844856

I. Introduction

RUL estimation has great significance for condition-based maintenance decision making [1], [2]. Since we are in the age of big data, massive data from sensors installed on prognostic and health management (PHM) systems can be easily available. However, how to effectively extract features from data and accurately predict RUL is still considered as a challenge for PHM [3]–[7].

RUL estimation methods could be coarsely classified as model-based and data-driven approaches. Model-based methods depend on building mathematical models to reflect the degradation trend of machines after the fault mechanism is fully understood. Then, based on collected data, the model parameters can be estimated [8]–[11]. However, establishing the physical model to describe the degradation trend depends on fully understanding the failure mechanism of the system. This kind of information is hard to obtain, especially when the system is complicated. Data-driven methods attempt to detect the degradation law based on the measured data. Statistical tools and machine learning are common data-driven methods to predict RUL. Si *et al.* [12] presented a comprehensive review of statistical data-driven approaches. Based on artificial neural network, Gebraeel *et al.* [13] built the data-driven model and estimated bearing RUL. Huang *et al.* [14] proposed a SOM, which treated time domain and frequency domain features as input to derive a novel HI called minimum quantization error, then back propagation neural networks were trained for degradation modeling. Chen *et al.* [15] combined adaptive neuro-fuzzy inference systems and high-order particle filtering for machine prognosis. Loutas *et al.* [16] presented SVR for bearing RUL prediction, which utilizes statistical features from multiple domains. Relevance vector machine was proposed to handle the limitation of

SVR in lacking probabilistic prediction and has been applied in the prognostics area [17], [18]. Gaussian process regression was proposed to estimate the degradation trend of slow speed bearing using acoustical emission signal [19]. More data-driven approaches can be found in a recent review paper [20]. Though data-driven methods are feasible and effective under many situations, they usually require manually proposing or extracting HIs by signal processing or statistical projection. Moreover, data-driven methods usually separate the task of feature extraction and model training such that great efforts in model selection and parameter tuning are required. It is highly demanding to develop automatic feature representation and learning approaches for RUL estimation.

Recently, deep learning (DL) has emerged and achieved much success in computer vision, machine translation, and social network filtering. DL attempts to learn high-level representations from the data through multiple layers. With the deep representation, DL possesses the strong capacities in building the relationship between degradation trend and measured data. Moreover, DL can learn a hierarchical feature representation automatically instead of designing hand-crafted features. In this paper, our focus is on a commonly known DL model, convolutional neural network (CNN), first proposed by LeCun *et al.* [21] for image classification.

Motivated by the strong power of CNN, researchers have done many related works about condition monitoring. To the authors' best knowledge, CNN for fault diagnosis has been widely studied, but prognosis work is relatively lacking. For diagnosis using two-dimensional (2-D) CNN, Janssens *et al.* [22] regarded discrete Fourier transform of two vibration signals as input for CNN and successfully classified the fault conditions of rotating machinery. Guo *et al.* [23] presented a new adaptive CNN, in which an adaptive learning rate was introduced to take both convergence time and loss error into consideration instead of a global constant learning rate. Liu *et al.* [24] added a dislocated layer before convolutional layer by considering the properties of industrial signal, thus revealed the relationship between signals with inconsistent impact intervals. Jing *et al.* [25] proposed an adaptive multisensor data fusion approach through a deep convolutional network for a gearbox fault diagnosis. Weimer *et al.* [26] performed a visual defect inspection by studying various design configurations of a deep CNN. Ince *et al.* [27] investigated a one-dimensional (1-D) CNN to perform fault diagnosis for a motor using raw time series. Abdeljaber *et al.* [28] further extended a 1-D CNN for vibration-based structural damage detection. As for prognosis related to CNN, Babu *et al.* [29] treated RUL prediction as multivariate time series regression and used a CNN-based regression approach. The effectiveness of the method was validated on a C-MAPSS data set.

To fill the research gap, we propose a new deep feature learning approach for RUL prediction, which relies on TFR and MSCNN. Because bearing degradation signal is complex and nonstationary, TFR can represent such kind of information effectively. Since we treat the RUL prediction as a regression problem, we try to find the relationship between the degraded TFR and RUL directly based on the MSCNN. After the degradation signals are collected, we utilize WT to get TFR of each sample. As the dimension of these TFRs is high, we employ a bilinear interpolation to reduce the size. Then, these resized TFRs together with their assigned RULs are sent to MSCNN to build the prediction model. Different from the traditional CNN structure that only utilizes features in the last convolutional layer, MSCNN structure integrates the last convolutional layer with the pooling layer before to form a multiscale (mixed) layer so that multiple levels of abstraction are kept for the prediction. The effectiveness of the proposed MSCNN method has been verified by the experimental data.

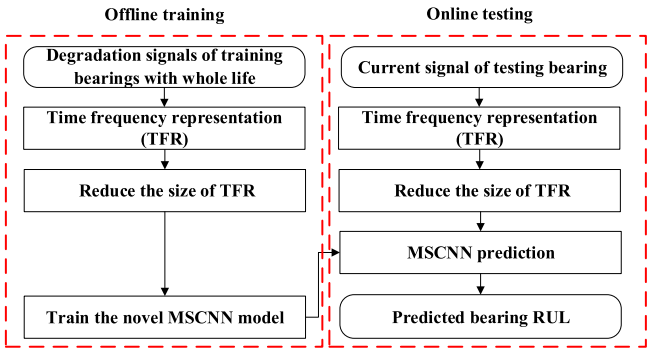


Fig. 1. Flowchart of the proposed method.

The contributions of this paper can be summarized as follows.

1) We employ an MSCNN model structure to extract more identifiable features for RUL prediction. By the multiscale layer, the global and local features are maintained to enhance the network capacity.
2) The proposed systematic approach integrates TFR and MSCNN into a framework, which could realize the goal of estimating RUL automatically.
3) To our best knowledge, this study first leverages MSCNN for bearing RUL prediction.

The rest of this paper is arranged as follows. Section II provides the theoretical background. Section III demonstrates the effectiveness of the presented RUL estimation approach using an experimental bearing data set. Section IV summarizes some discussions about the proposed method. Finally, the conclusion is given in Section V.

II. Theoretical Background

To address the limitations of traditional data-driven methods, the novel deep feature learning method combines TFR and MSCNN to automatically predict RUL. The flowchart of the proposed method is shown in Fig. 1. Our model assumption is that the health condition of the system degenerates linearly with usage. Since CNN is a supervised learning approach, we can assign target RUL value as the actual time left before failure based on this assumption [30]. The measured degradation signal contains rich useful information. However, there exists a nonstationary characteristic in raw time series, which causes difficulty in applying CNN directly. To represent the nonstationary property effectively, TFR by means of WT is applied for each degradation signal. Since these TFRs can be regarded as high-dimensional features, it is necessary to perform dimen-

sionality reduction. Instead of using common approaches such as principle component analysis (PCA), we introduce a simple and effective method, which reduces the size of TFR using bilinear interpolation. In this manner, the computation speed is increased. Then, resized TFRs together with their assigned RUL are sent to MSCNN to train the model. The multiscale layer in MSCNN keeps the global and local features in synch such that the characteristic of robust invariants and accurate details can contribute better for RUL estimation. In the training process, we tune the parameters by SGD and minimize MSE based on a BP algorithm. After the model is trained, we predict the RUL for each newly testing sample. The TFR for each sample can represent frequency energy changes with regard to the whole degradation process. The key for successfully predicting RUL lies in the effective feature mining in this model. The main techniques are briefly described as follows.

A. Time Frequency Representation

When the rotation machinery begins to degrade, the measured vibration signal will exhibit a nonstationary characteristic. Under this situation, both time domain and frequency domain analysis fail to provide valuable degradation information. Time-frequency analysis has been developed for nonstationary signals because it gives information both in time and frequency domain. Compared to short time Fourier transform, which has fixed time frequency resolution and Wigner–Ville distribution, which has the deficiency of the cross-term, WT is an effective TFR for nonstationary signal with a scalable resolution and does not have the problem of the cross-term. WT is widely applied in rotation machinery condition monitoring. Grossmann and Morlet [31] first put forward the novel concept of wavelet and formalized the continuous WT

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

where α is the scaling parameter, β is the translating parameter, $x(t)$ is a 1-D degradation signal, $\psi(t) \in L^2(R)$ is a mother wavelet function, and $\overline{\psi}(t)$ is the complex conjugate of $\psi(t)$. Since there is no standard or general method to select mother wavelet, we choose Morlet wavelet as the mother wavelet, which is similar to mechanical bearing impulse signal [32]. Through WT, 1-D degradation signal $x(t)$ is mapped to 2-D coefficients $U(\alpha,\beta)$, which is also called TFR here. We can identify a certain frequency (parameter α) at a particular time constant (parameter β).

B. Dimensionality Reduction

U can be regarded as high-dimensional feature, which contains plenty of degradation information. In order to reduce the computational burden for the subsequent MSCNN model, it is necessary to perform dimensionality reduction. Instead of using PCA, we employ a simple and valid method called bilinear interpolation, which is commonly used in image processing [33]

$$V = \phi(U) \tag{2}$$

where ϕ denotes the interpolation function, V is reduced low-dimensional feature.

C. Multiscale Convolutional Neural Network

The structure of traditional CNN generally includes an input layer, an output layer, and multiple hidden layers. The choices of hidden layers are convolutional, pooling, and fully connected. In a convolutional layer, local features are generated by convolutional kernels (to be learned in training process) because of sparse connectivity among neurons of adjacent layers. By means of a series of convolutions, the desired features can be rearranged and mined owing to similar statistical characteristic among the feature maps. A nonlinear activation function is imposed after convolution. The output feature map of convolutional layer can be written as

$$V_j^r = \varphi\left(\sum_i V_i^{r-1} * l_{ij}^r + b_j^r\right) \tag{3}$$

where $*$ means the convolution operator, V_i^{r-1} and V_j^r are the ith input feature map of layer $r-1$ and the jth output feature map of layer r in the convolution process. l_{ij}^r and b_j^r are the convolution kernel and bias. φ is a nonlinear activation function.

In the pooling layer, we take advantage of subsampling so that the output feature map becomes invariant to small variance in the input feature map. What is more, the computational efficiency is increased owing to the reduced size of feature map. Max-pooling is utilized, which is given as

$$V_j^{r+1}(c,d) = \max_{0\le p,q<m}\left\{V_j^r\left(c\cdot m+p, d\cdot m+q\right)\right\} \tag{4}$$

where V_j^r and V_j^{r+1} are the jth input feature map of layer r and the jth output feature map of layer $r+1$. We set the pooling filter size $m \times m$ to be 2×2.

The fully connected layer is usually added after several rounds of convolutional and pooling layer. Then, for classification problem, the soft-max layer is applied to classify the pattern of the data. However, because RUL estimation here is a regression problem, we employ the regression layer instead.

From the structure of traditional CNN, we know that the neurons in just one layer are regarded as final features for regression. In this way, the extracted global features (high-level) are much more invariant and stable than those in previous layers. However, some detailed local features (low-level) will be lost. To overcome this problem, we take advantages of MSCNN, which combines the last convolutional layer with the pooling layer before to form a mixed layer. Then, this mixed layer is followed by the fully connected (regression) layer. The global and local information can be kept synchronously under this manner to provide more precise regression results. The effectiveness of the MSCNN model structure for deep extracting global and local features for classification has already been proven in the literatures. Sun *et al.* [34] merged the last convolutional layer with

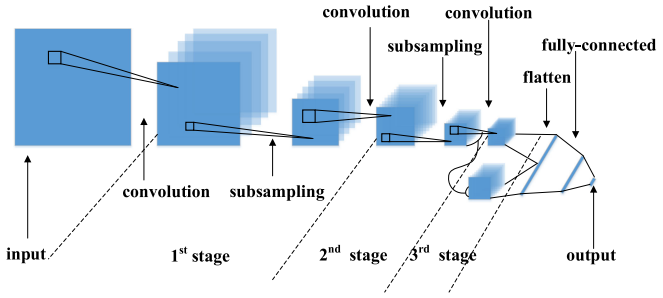


Fig. 2. MSCNN structure.

the pooling layer before to form a multiscale layer and gained promising performance in a face identification task. Ding and He [35] further employed this MSCNN and mined effective features for bearing fault diagnosis. The structure of MSCNN is shown in Fig. 2. The hidden layers contain three convolutional layers, two pooling layers and a mixed layer. The mixed layer accepts features, which are from unfolding the third convolutional layer and the second pooling layer. Hence, the output of the mixed layer is calculated as

$$V_{\text{final}} = \varphi\left(\sum_i f_i^1 \cdot w_{ij}^1 + \sum_i f_i^2 \cdot w_{ij}^2 + b_j\right) \tag{5}$$

where f_i^1, w_{ij}^1, f_i^2, and w_{ij}^2 mean the neurons and weights in the last convolutional layer and the second pooling layer, respectively. Finally, V_{final} is sent into the regression layer to perform RUL estimation. The loss function using MSE is defined as

$$L(\theta; V_{\text{final}}, y) = \frac{1}{2}||h_\theta(V_{\text{final}}) - y||^2 \tag{6}$$

where y is the ground truth output for RUL. h_θ is the regression function in the last layer for RUL prediction. θ denotes the parameters for the regression function. The rest of the parts of MSCNN are quite similar to the traditional CNN.

D. Parameters Optimization

The whole model parameter set Θ of MSCNN can be optimized by SGD to minimize loss function through BP algorithm. For a single sample $(x^{(k)}, y^{(k)})$, we can define square error as

$$L\left(\Theta; x^{(k)}, y^{(k)}\right) = \frac{1}{2}\left|\left|g_\Theta\left(x^{(k)}\right) - y^{(k)}\right|\right|^2 \tag{7}$$

where g_Θ is the complex function mapping from input data to output label. Assume we have the training data set $(x^{(1)}, y^{(1)}), \ldots, (x^{(N)}, y^{(N)})$ with N samples, then overall MSE is defined as

$$L(\Theta) = \frac{1}{N}\sum_{k=1}^{N} E\left(\Theta; x^{(k)}, y^{(k)}\right) + \frac{\lambda}{2}\sum_{r=1}^{n_r-1}\sum_{i=1}^{s_r}\sum_{j=1}^{s_{r+1}}\left(W_{ij}^r\right)^2 \tag{8}$$

where W_{ij}^r is the weight correlated with connection between the ith input of layer r and the jth output of layer $r+1$, s_r is the number of neurons in layer r, n_r is total number of layers, and λ denotes weight decay parameter. In (8), the first part stands for MSE, the second part acts as a regularization term. The goal of the BP is to minimize the contribution of the model parameters to $L(\Theta)$. By calculating the derivative of $L(\Theta)$ with respect to an individual weight and bias, we can perform SGD to minimize overall error in an iterative manner. The detailed explanation for this can be referred to [36].

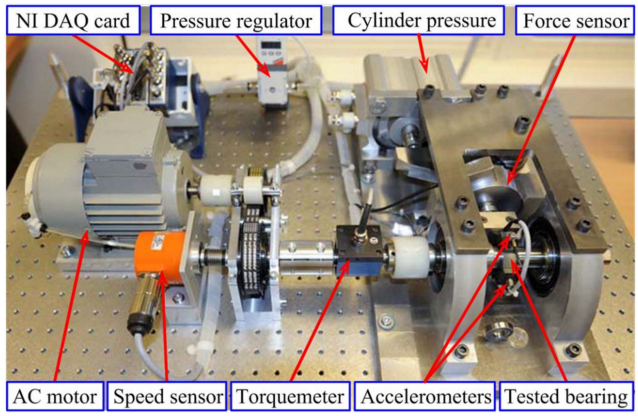


Fig. 3. Bearing experiment platform.

TABLE I
SPECIFICATIONS OF THE EXPERIMENT DATA

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

III. EXPERIMENTAL VERIFICATION

A. Experimental Description

The experimental data come from PRONOSTIA in the IEEE PHM 2012 Data Challenge [37]. The experiment platform is shown in Fig. 3. The platform mainly contains three major parts: a rotatory part, a degradation generation part, and a signal acquisition part. To accelerate the degradation of bearing, radial load force is applied with a controllable shaft speed. Two accelerated sensors perpendicular to each other are installed on the key position for the test bearing. The sampling frequency is 25 600 Hz. Each sample has a duration of 0.1 s, which means each sample has 2560 points. The record interval is 10 s. The test is ceased once the amplitude of the collected signal surpasses a certain level to prevent damage.

There are totally 17 run-to-failure data sets with three different operating conditions as shown in Table I.

B. Data Description

Our proposed method is applied on the data set in the operation condition 1, which means that our method currently only considers the operation condition with constant speed and load. The run-to-failure bearing1_1 and bearing1_2 are the training data sets, which contain 2803 samples and 871 samples, respectively. The other bearings in this operating condition are

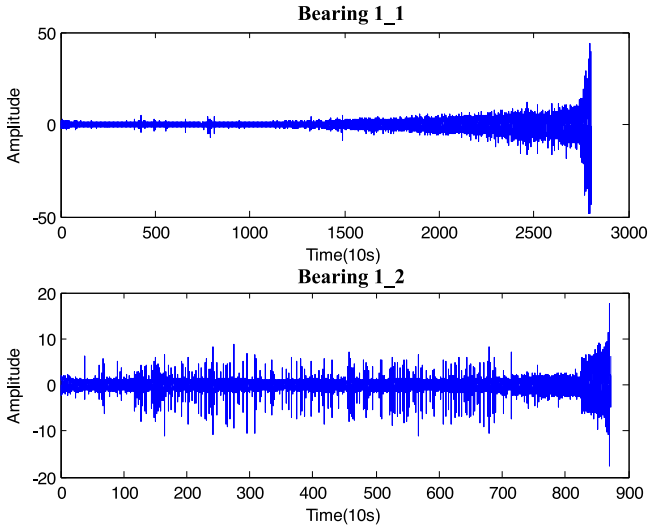


Fig. 4. Whole lifetime vibration signal from horizontal direction.

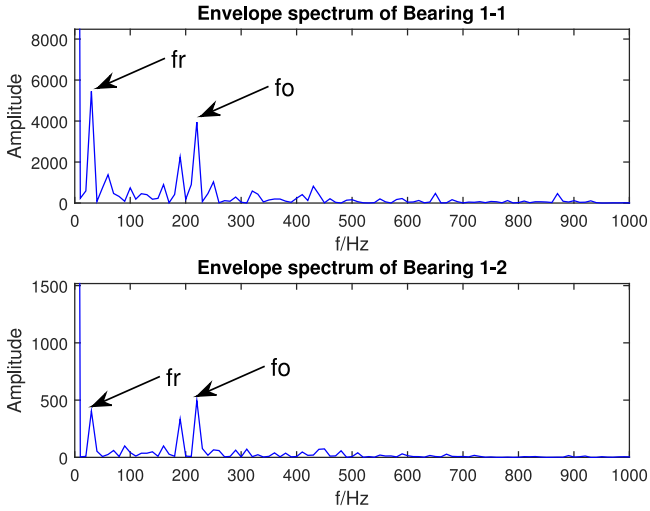


Fig. 5. Envelope spectrum of the last sample in the training data.

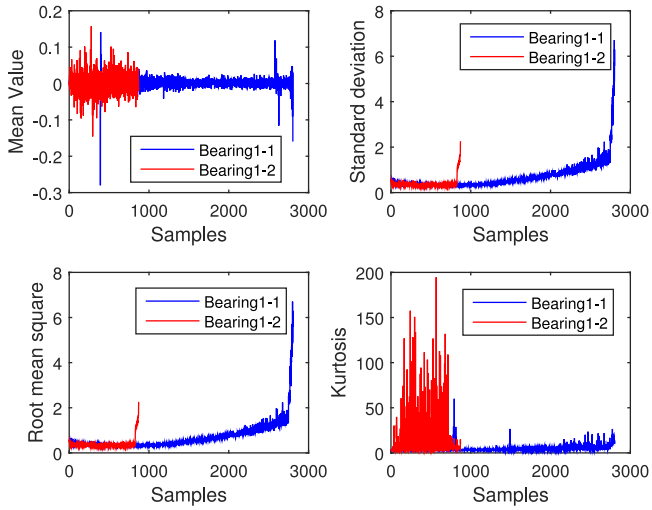


Fig. 6. Basic statistic description of training data.

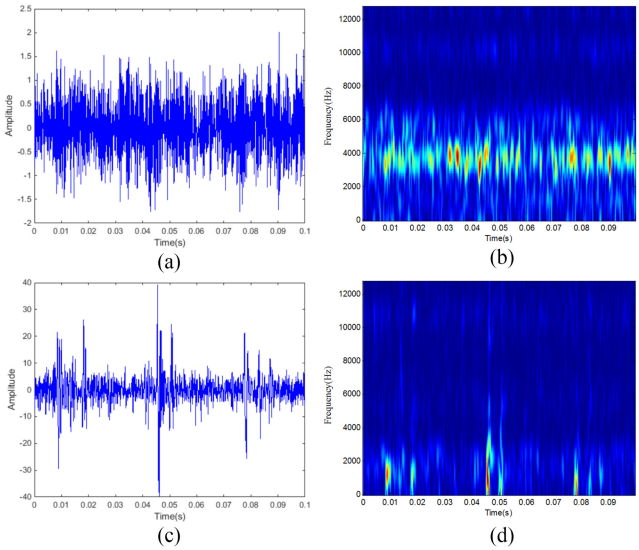


Fig. 7. Bearing 1_1: (a) Waveform for the first sample; (b) TFR for the first sample; (c) waveform for the last sample; and (d) TFR for the last sample.

regarded as the testing data sets with censored bearing life data. In this study, the measured signals from both horizontal and vertical direction are considered to give more comprehensive information for bearing degradation process. Under this situation, the number of input channels for MSCNN is two. To see how the raw signal in time changes over time, we plot the lifetime data from horizontal direction for bearing1_1 and bearing1_2 as shown in Fig. 4. Bearing1_1 shows gradual increasing trend, which suggests that the fault becomes severe gradually, while the amplitude of bearing1_2 demonstrates sudden increase near the end of lifetime so that abrupt degradation process is exhibited. We plot the envelope spectrum of the last sample for training data in Fig. 5. From the result, we can clearly identify the rotation frequency f_r and bearing fault characteristic frequency f_0. Meanwhile, to give the basic statistic description of training data, we also plot how the mean, standard deviation, root mean square, and kurtosis change with regard to each sample from the horizontal direction in Fig. 6. The mean value and kurtosis exhibit no regular pattern. The standard deviation and root mean square increase gradually as the fault develops.

C. RUL Prediction the Using Proposed Method

To implement the proposed method, we first transform all training samples into TFRs using WT. To illustrate the frequency energy change with regard to time, TFR of the first and last sample in bearing1_1 and bearing1_2 are given in Figs. 7 and 8. From the plot, we can see that the bearings are in the normal stage at the beginning, which the rotation frequency manifests and frequency fluctuation is not obvious in TFR. However, at the end, the bearing with a defect condition will show phenomenon of the periodic impulse, which causes the effect of modulation. These TFRs (704×2560) are resized to 30×30 to form a standard image size using bilinear interpolation. Then, these resized TFRs together with their assigned RUL are used for

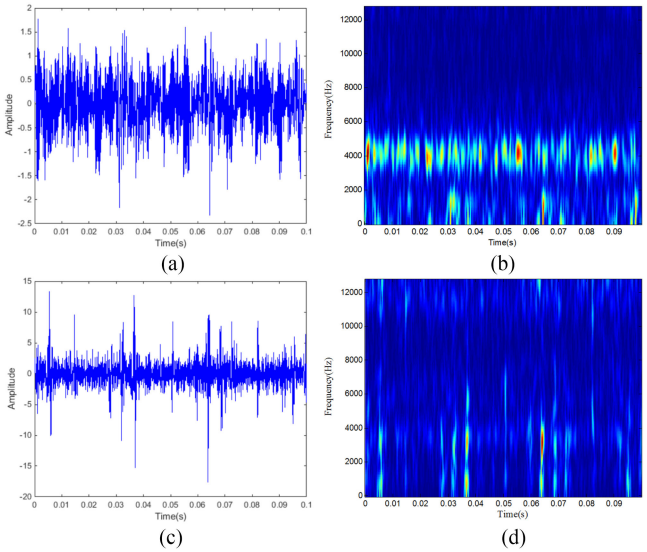


Fig. 8. Bearing 1_2: (a) Waveform for the first sample; (b) TFR for the first sample; (c) Waveform for the last sample; and (d) TFR for the last sample.

TABLE II
TOPOLOGY OF MSCNN

Layer	Parameters and output channel size
input	size:30×30, channel:2
convolution	kernel:3×3, channel:12
pooling	kernel:2×2, stride:2, channel:12
convolution	kernel:5×5, channel:24
pooling	kernel:2×2, stride:2, channel:24
convolution	kernel:3×3, channel:24
multiscale	channel:816(216+600)
fully-connected	channel:200
output	channel:1

training the MSCNN model. After the MSCNN model is built, we can predict RUL for any sample in the testing data set. The topology of the MSCNN is shown in Table II, which has two input TFRs with a size of 30 × 30, a convolutional layer with 12 filters with a size of 3 × 3, a subsampling layer with a pooling size of 2 × 2 and stride 2, a convolutional layer with 24 filters with a size of 5 × 5, a subsampling layer with a pooling size of 2 × 2 and stride 2, a convolutional layer with 24 filters with a size of 3 × 3, a multiscale layer with 816 hidden units, a fully connected layer with 200 hidden units, and an output layer with 1 unit. The rectified linear units (ReLU) activation function is used for the whole model, which can relieve gradient vanishing or explosion problem. The ReLU activation function is given as

$$\varphi(z) = \begin{cases} 0, & \text{if} \quad z < 0 \\ z, & \text{otherwise} \end{cases}. \tag{9}$$

For the training of MSCNN, the weights for all layers are initialized with mean 0 and standard deviation 0.01. The biases are initialized with 0. For the selection of hyperparameters such as learning rate, batch size, and weight decay, we tune these hyperparameters to get satisfactory training performance. Once the model is built, we can predict RUL for the testing data set.

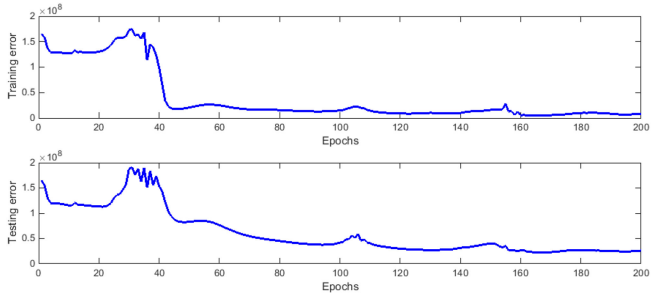


Fig. 9. Training and testing error over iterations.

Fig. 9 shows the training and testing error over iterations. We can see that after 200 iterations the error becomes somewhat stable and does not show much variation. Testing MSE is slightly lower than training MSE. Thus, we can say our built model is not overfitted. The reason lies in that we have introduced ReLU activation function and regularization techniques. Moreover, MSCNN has the skipping connection between layers so that backward gradient can be flowed easily. The training time is nearly half hour on a desk computer (64 bit, 3.30 GHz CPU, and 8 GB memory). The RUL prediction results for the testing data set are shown in Fig. 10. From the plot, we can see that the proposed method is effective in detecting the degradation trend for most of bearings and estimating RUL. It does not perform well on bearing1_5 as the proposed method fails to get the degradation trend during its censored life data. As mentioned in [38], the reason may be multiple faults existing in bearing1_5. Therefore, to comprehensively assess the performance of the estimation approach, a score function in the IEEE PHM 2012 challenge [37] is used

$$\text{Score} = \frac{1}{5}\sum_{i=1}^{5} A_i \tag{10}$$

where

$$A_i = \begin{cases} \exp(-\ln(0.5)\cdot(Er_i/5)), & Er_i \le 0 \\ \exp(+\ln(0.5)\cdot(Er_i/20)), & Er_i > 0 \end{cases} \tag{11}$$

and Er_i denotes the percent error for the ith testing data set

$$Er_i = \frac{\text{ActRUL}_i - \hat{\text{RUL}}_i}{\text{ActRUL}_i} \times 100 \tag{12}$$

where ActRUL_i and $\hat{\text{RUL}}_i$ mean the actual RUL and the estimated RUL for the ith testing data set. In additional to the score function, performance metrics such as MAE and NRMSE are also included to make a further comparison

$$\text{MAE} = \frac{1}{5}\sum_{i=1}^{5}\left|\text{ActRUL}_i - \hat{\text{RUL}}_i\right| \tag{13}$$

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{5}\sum_{i=1}^{5}\left(\text{ActRUL}_i - \hat{\text{RUL}}_i\right)^2}}{\left(\frac{1}{5}\sum_{i=1}^{5}\hat{\text{RUL}}_i\right)}. \tag{14}$$

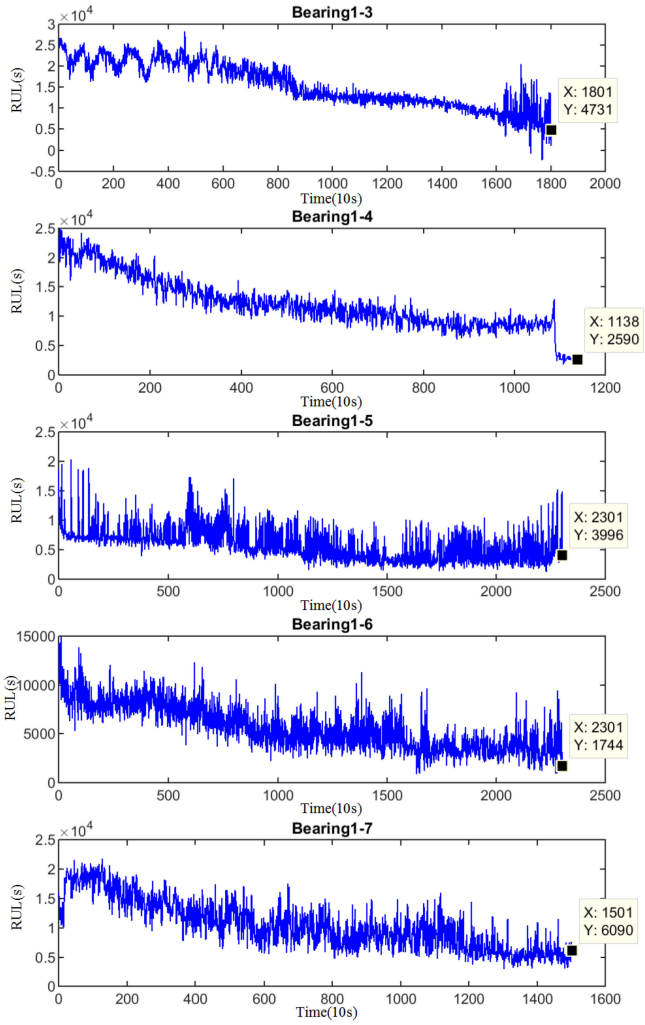


Fig. 10. Bearing RUL prediction.

The RUL estimation results for five testing bearings are shown in Table III.

D. Comparison With Other Data-Driven Methods

For comparisons, three data-driven based HI methods are also applied. The first data-driven method [39] relies on RNN to construct HI from a set of processed features. The RNN-HI possesses the properties of high monotonicity and correlation, which facilitates RUL prediction. The second data-driven method SOM-HI [38] is similar to the first data-driven method, in which HI is constructed by SOM. The third data-driven method [16] is based on SVR-HI. From Table III, we can see that the proposed method acquires the highest score and the lowest error among these methods, which suggests our proposed method provides more accurate RUL estimation results. Different from data driven based HI methods, which separate feature extraction and estimation, our deep feature learning model integrates them as a unified task. Under this manner, the effort for model selection and parameter tuning is greatly reduced and the prediction accuracy is increased.

TABLE III
RUL ESTIMATION RESULTS AND COMPARISON I

Testing dataset	Current time(s)	Actual RUL(s)	MSCNN $\hat{R}UL$(s)	RNN-HI $\hat{R}UL$(s)	SOM-HI $\hat{R}UL$(s)	SVR-HI $\hat{R}UL$(s)
Bearing1_3	18010	5730	4731	3250	5790	5970
Bearing1_4	11380	2900	2590	1100	410	1200
Bearing1_5	23010	1610	3996	1980	6080	5040
Bearing1_6	23010	1460	1744	1150	1180	1230
Bearing1_7	15010	7570	6090	6220	8110	9120
Score			0.3624	0.2798	0.3605	0.2657
MAE			1091.8	1262	1568	1430
NRMSE			0.3514	0.5522	0.5342	0.4107

TABLE IV
RUL ESTIMATION RESULTS AND COMPARISON II

Testing dataset	CNN $\hat{R}UL$(s)	Structure I_1 $\hat{R}UL$(s)	Structure I_2 $\hat{R}UL$(s)	Structure I_3 $\hat{R}UL$(s)	Structure II_1 $\hat{R}UL$(s)	Structure II_2 $\hat{R}UL$(s)
Bearing1_3	4492	4653	4536	4031	4683	4715
Bearing1_4	1220	2341	2147	1885	2438	2497
Bearing1_5	4501	4125	4328	4413	4237	4436
Bearing1_6	1759	1832	2035	2247	1848	1762
Bearing1_7	8217	5934	5832	5968	6035	5879
Score	0.1943	0.3072	0.2696	0.2272	0.3254	0.3354
MAE	1351	1231.8	1395.4	1581.2	1211.8	1247.4
NRMSE	0.4027	0.3857	0.4226	0.4664	0.3805	0.4037

E. Comparison With Other CNN-Based Feature Extraction Methods

To further illustrate the advantage of the proposed MSCNN method, traditional CNN and other CNN-based feature extraction methods are introduced. These methods have the similar structure as the MSCNN except the size of the multiscale layer. The traditional CNN [40] utilizing the basic model structure without the multiscale layer is first compared. Then, we change the CNN structure to form the multiscale layer by merging different layers. We denote the first convolutional layer as conv1, the first pooling layer as pool1, the second convolutional layer as conv2, the second pooling layer as pool2, and the third convolutional layer as conv3. Conv3+conv2, conv3+pool1, and conv3+conv1 are regarded as structure I_1, I_2, and I_3. What is more, conv3+conv2+conv1 and conv3+pool2+pool1 are considered as structure II_1 and II_2. The structures I and II are presented to further illustrate the advantage of the proposed MSCNN structure. From Table IV, the proposed method achieves the best performance among all the methods. Because of the multiscale structure, MSCNN has obvious advantage compared to traditional CNN. Combining more features does not contribute much for RUL prediction since the included features are redundant to some degree. Moreover, the structures I and II require more effects in regularization to avoid overfitting and the computation time is longer.

IV. DISCUSSIONS

The effectiveness and advantage of the proposed MSCNN method have been verified by experimental data, and there are also some points that need to be indicated, which are future research directions.

1) We get TFR of each sample by using WT. In this paper, we predefine the mother wavelet as Morlet wavelet based on experience since the property of Morlet wavelet matches well with defect bearing signal. However, according to Heisenberg–Gabor inequality, WT cannot keep a fine resolution in both time and frequency domains. Therefore, more advanced time-frequency transform techniques such as synchrosqueezing transform and

parametric time-frequency transform can improve RUL estimation performance.

2) To increase the computational speed, the dimension of the TFR matrix is reduced by bilinear interpolation. Though this method is simple and effective, some information contained in original TFR may be lost in this process and cause some errors. To overcome this issue, a graphic processing unit can help process the original TFR owing to its excellent computation performance. Moreover, we can parallel model structure to allow for fast computation and real-time operation.
3) MSCNN structure used in this paper keeps global robust and local precise features simultaneously so that it manifests better capacity in feature mining compared to traditional CNN. Although we have compared different CNN-based feature extraction methods, we still do not investigate how to form an optimal combination of layers for the MSCNN structure. More understanding about this will be made in the future.
4) Because overfitting is the common problem for neural network, we employ ReLU activation and regularization to relieve this problem. What is more, the MSCNN structure itself possesses the characteristics of skipping connection. More techniques such as dropout technique and sparse technique can be introduced to further relieve this problem.
5) Currently, the operation condition is assumed constant for our proposed method. We will extend our method to predicting RUL under different working conditions by considering the operation condition.
6) RUL estimation together with confidence interval could help engineers make maintenance decisions more comprehensively. To relieve the inherent limitation of the proposed method in failing to consider uncertainties, we will consider more bootstrap methods to construct the confidence bounds in the future [41]–[43].

V. Conclusion

In this paper, we utilized the MSCNN for bearing RUL prediction, which is the first attempt in this field. To settle the drawbacks of traditional data-driven methods, we presented a deep feature learning method combining TFR and MSCNN, which can fulfill the task of RUL estimation automatically. TFRs of the degradation signal were obtained through WT. To increase the computational efficiency, we employed bilinear interpolation to reduce the dimension of TFRs. Then, TFRs with reduced size were regarded as input for MSCNN. Owing to the multiscale layer in the MSCNN, which retains the global and local feature synchronously, the extracted features are especially salient and suitable for RUL prediction. After applying the proposed method to the experimental bearing data, the RUL estimation results proved the effectiveness of the proposed method. Compared to three traditional data-driven and several CNN-based feature methods, the proposed method achieved the best prediction accuracy and the lowest prediction error, which demonstrates the superiority of the proposed method. Moreover, further work directions were discussed.

References

[1] N. Chen, Z.-S. Ye, Y. Xiang, and L. Zhang, "Condition-based maintenance using the inverse Gaussian degradation model," *Eur. J. Oper. Res.*, vol. 243, no. 1, pp. 190–199, 2015.

[2] Z. Ye, N. Chen, and K.-L. Tsui, "A Bayesian approach to condition monitoring with imperfect inspections," *Quality Rel. Eng. Int.*, vol. 31, no. 3, pp. 513–522, 2015.

[3] N. Chen and K. L. Tsui, "Condition monitoring and remaining useful life prediction using degradation signals: Revisited," *IIE Trans.*, vol. 45, no. 9, pp. 939–952, 2013.

[4] D. Wang and K.-L. Tsui, "Statistical modeling of bearing degradation signals," *IEEE Trans. Rel.*, vol. 66, no. 4, pp. 1331–1344, Dec. 2017.

[5] D. Wang and K.-L. Tsui, "Two novel mixed effects models for prognostics of rolling element bearings," *Mech. Syst. Signal Process.*, vol. 99, pp. 1–13, 2018.

[6] D. Wang and K.-L. Tsui, "Brownian motion with adaptive drift for remaining useful life prediction: Revisited," *Mech. Syst. Signal Process.*, vol. 99, pp. 691–701, 2018.

[7] Q. Zhai and Z.-S. Ye, "RUL prediction of deteriorating products using an adaptive Wiener process model," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 2911–2921, Dec. 2017.

[8] Y.-S. Shih and J.-J. Chen, "Analysis of fatigue crack growth on a cracked shaft," *Int. J. Fatigue*, vol. 19, no. 6, pp. 477–486, 1997.

[9] Y. Li, S. Billington, C. Zhang, T. Kurfess, S. Danyluk, and S. Liang, "Adaptive prognostics for rolling element bearing condition," *Mech. Syst. Signal Process.*, vol. 13, no. 1, pp. 103–113, 1999.

[10] N. Z. Gebraeel, M. A. Lawley, R. Li, and J. K. Ryan, "Residual-life distributions from component degradation signals: A Bayesian approach," *IIE Trans.*, vol. 37, no. 6, pp. 543–557, 2005.

[11] Z. Tian and H. Liao, "Condition based maintenance optimization for multi-component systems using proportional hazards model," *Rel. Eng. Syst. Safety*, vol. 96, no. 5, pp. 581–589, 2011.

[12] X.-S. Si, W. Wang, C.-H. Hu, and D.-H. Zhou, "Remaining useful life estimation—A review on the statistical data driven approaches," *Eur. J. Oper. Res.*, vol. 213, no. 1, pp. 1–14, 2011.

[13] N. Gebraeel, M. Lawley, R. Liu, and V. Parmeshwaran, "Residual life predictions from vibration-based degradation signals: A neural network approach," *IEEE Trans. Ind. Electron.*, vol. 51, no. 3, pp. 694–700, Jun. 2004.

[14] R. Huang, L. Xi, X. Li, C. R. Liu, H. Qiu, and J. Lee, "Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods," *Mech. Syst. Signal Process.*, vol. 21, no. 1, pp. 193–207, 2007.

[15] C. Chen, B. Zhang, G. Vachtsevanos, and M. Orchard, "Machine condition prediction based on adaptive neuro–fuzzy and high-order particle filtering," *IEEE Trans. Ind. Electron.*, vol. 58, no. 9, pp. 4353–4364, Sep. 2011.

[16] T. H. Loutas, D. Roulias, and G. Georgoulas, "Remaining useful life estimation in rolling bearings utilizing data-driven probabilistic e-support vectors regression," *IEEE Trans. Rel.*, vol. 62, no. 4, pp. 821–832, Dec. 2013.

[17] P. Wang, B. D. Youn, and C. Hu, "A generic probabilistic framework for structural health prognostics and uncertainty management," *Mech. Syst. Signal Process.*, vol. 28, pp. 622–637, 2012.

[18] A. Widodo and B.-S. Yang, "Application of relevance vector machine and survival probability to machine degradation assessment," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2592–2599, 2011.

[19] S. Aye and P. Heyns, "An integrated Gaussian process regression for prediction of remaining useful life of slow speed bearings based on acoustic emission," *Mech. Syst. Signal Process.*, vol. 84, pp. 485–498, 2017.

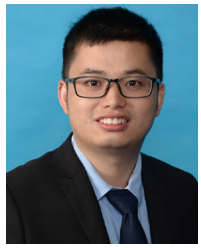
[20] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: A systematic review from data acquisition to RUL prediction," *Mech. Syst. Signal Process.*, vol. 104, pp. 799–834, 2018.

[21] Y. LeCun *et al.*, "Handwritten digit recognition with a back-propagation network," in *Proc. Adv. Neural Inf. Process. Syst.*, 1990, pp. 396–404.

[22] O. Janssens *et al.*, "Convolutional neural network based fault detection for rotating machinery," *J. Sound Vibration*, vol. 377, pp. 331–345, 2016.

[23] X. Guo, L. Chen, and C. Shen, "Hierarchical adaptive deep convolution neural network and its application to bearing fault diagnosis," *Measurement*, vol. 93, pp. 490–502, 2016.

[24] R. Liu, G. Meng, B. Yang, C. Sun, and X. Chen, "Dislocated time series convolutional neural architecture: An intelligent fault diagnosis approach for electric machine," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1310–1320, Jun. 2017.
[25] L. Jing, T. Wang, M. Zhao, and P. Wang, "An adaptive multi-sensor data fusion method based on deep convolutional neural networks for fault diagnosis of planetary gearbox," *Sensors*, vol. 17, no. 2, 2017, Art. no. 414.
[26] D. Weimer, B. Scholz-Reiter, and M. Shpitalni, "Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection," *CIRP Ann.-Manufacturing Technol.*, vol. 65, no. 1, pp. 417–420, 2016.
[27] T. Ince, S. Kiranyaz, L. Eren, M. Askar, and M. Gabbouj, "Real-time motor fault detection by 1-D convolutional neural networks," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 7067–7075, Nov. 2016.
[28] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman, "Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks," *J. Sound Vibration*, vol. 388, pp. 154–170, 2017.
[29] G. S. Babu, P. Zhao, and X.-L. Li, "Deep convolutional neural network based regression approach for estimation of remaining useful life," in *Proc. Int. Conf. Database Syst. Adv. Appl.*, 2016, pp. 214–228.
[30] L. Peel, "Data driven prognostics using a Kalman filter ensemble of neural network models," in *Proc. Int. Conf. Prognostics Health Manag.*, 2008, pp. 1–6.
[31] A. Grossmann and J. Morlet, "Decomposition of Hardy functions into square integrable wavelets of constant shape," *SIAM J. Math. Anal.*, vol. 15, no. 4, pp. 723–736, 1984.
[32] B. Tang, W. Liu, and T. Song, "Wind turbine fault diagnosis based on Morlet wavelet transformation and Wigner-Ville distribution," *Renew. Energy*, vol. 35, no. 12, pp. 2862–2866, 2010.
[33] H. Raveendran and D. Thomas, "Image fusion using LEP filtering and bilinear interpolation," *Int. J. Eng. Trends Technol.*, vol. 12, no. 9, pp. 427–431, 2014.
[34] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10 000 classes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1891–1898.
[35] X. Ding and Q. He, "Energy-fluctuated multiscale feature learning with deep convnet for intelligent spindle bearing fault diagnosis," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 8, pp. 1926–1935, Aug. 2017.
[36] J. Bouvrie, "Notes on convolutional neural networks," Center Biol. Comput. Learn., Massachusetts Inst. Technol., Cambridge, MA, USA, MIT CBCL Tech Rep., pp. 38–44, 2006.
[37] P. Nectoux *et al.*, "Pronostia: An experimental platform for bearings accelerated degradation tests," in *Proc. IEEE Int. Conf. Prognostics Health Manag.*, 2012, pp. 1–8.
[38] 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 Process.*, vol. 27, pp. 159–166, 2014.
[39] 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*, vol. 240, pp. 98–109, 2017.
[40] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[41] B. Efron and G. Gong, "A leisurely look at the bootstrap, the jackknife, and cross-validation," *Amer. Statist.*, vol. 37, no. 1, pp. 36–48, 1983.
[42] C. Sbarufatti, M. Corbetta, A. Manes, and M. Giglio, "Sequential Monte-Carlo sampling based on a committee of artificial neural networks for posterior state estimation and residual lifetime prediction," *Int. J. Fatigue*, vol. 83, pp. 10–23, 2016.
[43] J. Deutsch and D. He, "Using deep learning-based approach to predict remaining useful life of rotating components," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 48, no. 1, pp. 11–20, Jan. 2018.



Jun Zhu received the B.S. degree in mechanical design manufacture and automation from Huazhong Agricultural University, Wuhan, China, in 2013, and the M.S. degree in mechatronic engineering from the University of Science and Technology of China, Hefei, China, in 2016. He is currently working toward the Ph.D. degree in industrial engineering with the National University of Singapore, Singapore.

His research interests include signal processing and data mining for machinery prognostics and health management.

Nan Chen (M'11) received the B.S. degree in automation from Tsinghua University, Beijing, China, in 2006, the M.S. degree in computer science in 2009, and the M.S. degree in statistics and the Ph.D. degree in industrial engineering from the University of Wisconsin-Madison, Madison, WI, USA, both in 2010.

He is currently an Associate Professor with the Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore. His research interests include statistical modeling and surveillance of engineering systems, simulation modeling design, condition monitoring, and degradation modeling.

Weiwen Peng received the B.S. degree in mechanical design manufacture and automation, the M.S. degree in mechanical design and theory, and the Ph.D. degree in mechanical engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2009, 2012, and 2015, respectively.

He is currently a Research Fellow with the Department of Industrial Systems Engineering and Management, National University of Singapore, Singapore. His research interests include degradation modeling, machinery prognostics, and Bayesian machine learning in reliability engineering.