


Article

# A Novel Method for Early Gear Pitting Fault Diagnosis Using Stacked SAE and GBRBM

Jialin Li <sup>1</sup>, Xueyi Li <sup>1</sup>, David He <sup>1,2,\*</sup> and Yongzhi Qu <sup>3</sup>

<sup>1</sup> School of Mechanical Engineering and Automation, Northeastern University, Shenyang 110000, China; jialinli\_neu@163.com (J.L.); lixueyineu@gmail.com (X.L.)

<sup>2</sup> Department of Mechanical and Industrial Engineering, The University of Illinois at Chicago, Chicago, IL 60607, USA

<sup>3</sup> School of Mechanical and Electronic Engineering, Wuhan University of Technology, Wuhan 430000, China; yongzhiqu@hotmail.com

\* Correspondence: davidhe@uic.edu; Tel.: +86-132-6253-3830

Received: 16 January 2019; Accepted: 9 February 2019; Published: 13 February 2019



**Abstract:** Research on data-driven fault diagnosis methods has received much attention in recent years. The deep belief network (DBN) is a commonly used deep learning method for fault diagnosis. In the past, when people used DBN to diagnose gear pitting faults, it was found that the diagnosis result was not good with continuous time domain vibration signals as direct inputs into DBN. Therefore, most researchers extracted features from time domain vibration signals as inputs into DBN. However, it is desirable to use raw vibration signals as direct inputs to achieve good fault diagnosis results. Therefore, this paper proposes a novel method by stacking sparse autoencoder (SAE) and Gauss-Binary restricted Boltzmann machine (GBRBM) for early gear pitting faults diagnosis with raw vibration signals as direct inputs. The SAE layer is used to compress the raw vibration data and the GBRBM layer is used to effectively process continuous time domain vibration signals. Vibration signals of seven early gear pitting faults collected from a gear test rig are used to validate the proposed method. The validation results show that the proposed method maintains a good diagnosis performance under different working conditions and gives higher diagnosis accuracy compared to other traditional methods.

**Keywords:** early gear pitting fault diagnosis; vibration signals; SAE; GBRBM

## 1. Introduction

Gears play an important role in mechanical transmission systems. It is necessary to diagnose gear faults to ensure stable and reliable operation of the systems. The methods of fault diagnosis can be roughly divided into two categories: model-driven methods and data-driven methods [1]. Model-based diagnostic methods require a deep understanding of the systems, and many parameter adjustments need to be performed to build the model. Therefore, this paper applies data-driven methods to diagnose gear faults. The data-driven diagnostic process involves two steps: (1) establish a data model based on known state data, (2) use the established model to diagnose mechanical faults. The fault diagnosis process can be regarded as the process of applying the model for pattern recognition. When building a fault diagnosis model, there are generally two processes: feature extraction and pattern recognition [2]. The purpose of feature extraction is to convert high-dimensional data into low-dimensional features, which can better perform pattern recognition. There are many methods for feature extraction such as statistical analysis methods, fast Fourier transform (FFT), Hilbert–Huang transform (HHT) [3], empirical mode decomposition (EMD) [4], wavelet transform (WT) [5], principal components analysis (PCA) [6], and so on. There are many traditional pattern recognition methods

including Bayesian classifier [7], K-nearest neighbor (KNN) algorithms [8], artificial neural network (ANN), support vector machine (SVM) [9], etc. Traditional ANN can only distinguish less complex features, and the diagnosis results are greatly affected by process of feature extraction and feature selection. In recent years, research on deep learning is getting popular. Deep learning can improve the shortcomings of traditional neural network. There are many types of deep learning methods applied in fault diagnosis, which can be divided into supervised methods such as deep neural network (DNN), convolutional neural network (CNN) [10] and unsupervised methods such as deep belief network (DBN), and autoencoder (AE).

Zhang et al. [11] used DNN to diagnose bearing faults and directly used the collected vibration signals as the inputs of neural network, removing the error caused by the feature extraction process. Tested by two publicly available data from University of Cincinnati Center for Intelligent Maintenance System (IMS) and Case Western Reserve University (CWRU), their proposed method was shown to be able to effectively diagnose bearing faults. Chen et al. [12] used a CNN to diagnose gearbox faults. FFT was performed on the vibration signals. Statistical methods were used to extract features from the time-domain and the frequency domain signals as inputs to the CNN. Chen et al. [13] applied ensemble empirical mode decomposition (EEMD) to extract features, and then applied DBN to classify the gear faults. Wang et al. [14] applied unsupervised continuous sparse autoencoder (CSAE) for feature learning and connected a layer of back propagation (BP) networks behind the CSAE. The training process first applied CSAE for unsupervised classification, and then used BP for supervised learning.

DBN has been used for fault diagnosis. The research using DBN for fault diagnosis is reviewed next. Tran et al. [15] used Teager–Kaiser energy operator (TKEO) and DBN to diagnose reciprocating compressor valves faults. In their paper, TKEO was proposed to estimate the amplitude envelopes. The collected vibration signal was processed by WT denoising, and then the time domain signal was converted into a frequency domain. Finally, the statistical methods were used to extract the feature as inputs of the DBN. The diagnostic method used by Han et al. [16] was similar to the method in reference [15], except that it adds a particle swarm optimization-support vector machine (PSO-SVM) to classify extracted parameters. Shao et al. [17] applied the dual-tree complex wavelet packet for feature extraction, and then used statistical methods for feature selection. Finally, an adaptive DBN was used for fault classification. Wang et al. [18] also applied statistical methods to process time-frequency domain signals, and then used DBN to detect multiple faults in axial piston pumps. Lee et al. [19] used DBN to diagnose the air handling unit (AHU). Ahmed et al. [20] combined DBN and softmax classifiers to diagnose rolling bearing faults. Tao et al. [21], He et al. [22], and Chen et al. [23] also applied statistical methods for feature extraction, and then applied DBN for fault classification.

Deutsch et al. [24] integrates DBN and a particle filter for bearing remaining useful life (RUL) prediction. Geng et al. [25] were inspired by the glial chains to improve the structure of the restricted Boltzmann machines (RBMs). An improved greedy layer-wise learning algorithm was used to improve the diagnostic accuracy. Ren et al. [26] combined deep belief networks and multiple models (DBN-MMs) to diagnose complex systems faults. Shao et al. [27] combined the CNN with the DBN to process the compressed sensing (CS). In addition, exponential moving average (EMA) technique was used to improve diagnostic accuracy of the constructed deep model. Jiang et al. [28] proposed a feature fusion DBN method to diagnose rotating machinery fault. Moreover, the locality preserving projection (LPP) was used to fusion deep features to further improve the quality of the deep features.

SAE has been used for fault diagnosis recently. The research using SAE to diagnose faults is reviewed next. Shao et al. [29] proposed ensemble deep auto-encoders (EDAEs) to diagnose bearing faults. The effects of different activation functions and various AEs on diagnostic results were discussed in the article. Maurya et al. [30] used stacked autoencoder to fuse the low-level feature. And then a multi-class SVM was used as classifier. Shao et al. [31] applied deep autoencoder to diagnose rotating machinery faults. The maximum cross entropy was used as the loss function and artificial fish swarm (AFS) algorithm was applied to optimize the key parameters of the deep autoencoder. Meng et al. [32] used denoising autoencoder to diagnose bearing faults. They improved the fault diagnosis rate by

reusing the data points between the adjacent samples. The hyper parameter was adjusted by changing the number of units per layer to adapt to the different resilience of the layer.

This paper proposed integrates SAE with GBRBM to diagnose early gear pitting faults. The SAE is used to convert high-dimensional data into low-dimensional data, and GBRBM is used to accommodate the continuous distribution of the inputs. The rest of this paper is organized as follows. In Section 2, the proposed method based the stacked SAE and GBRBM is explained in details. In Section 3, the description of the experimental test rig used for collecting the vibration data for the seven gear pitting faults is provided. In Section 4, the validation results and the discussion of the validation results are presented. Finally, Section 5 concludes the paper.

## 2. The Proposed Method

### 2.1. Framework of Proposed Method

Most of the data-driven diagnosis methods involve separate manual feature extraction process. Manual feature extraction mostly relies on human expertise, and the manual feature extraction process is time-consuming and labor intensive. Moreover, the diagnostic results are greatly affected by the feature extraction method. Therefore, diagnostic methods that do not include separate manual feature process are more desirable. Inspired by the unsupervised learning process, this paper proposes a diagnostic method that combines supervised learning with unsupervised learning. The framework of the diagnostic method is shown in Figure 1.

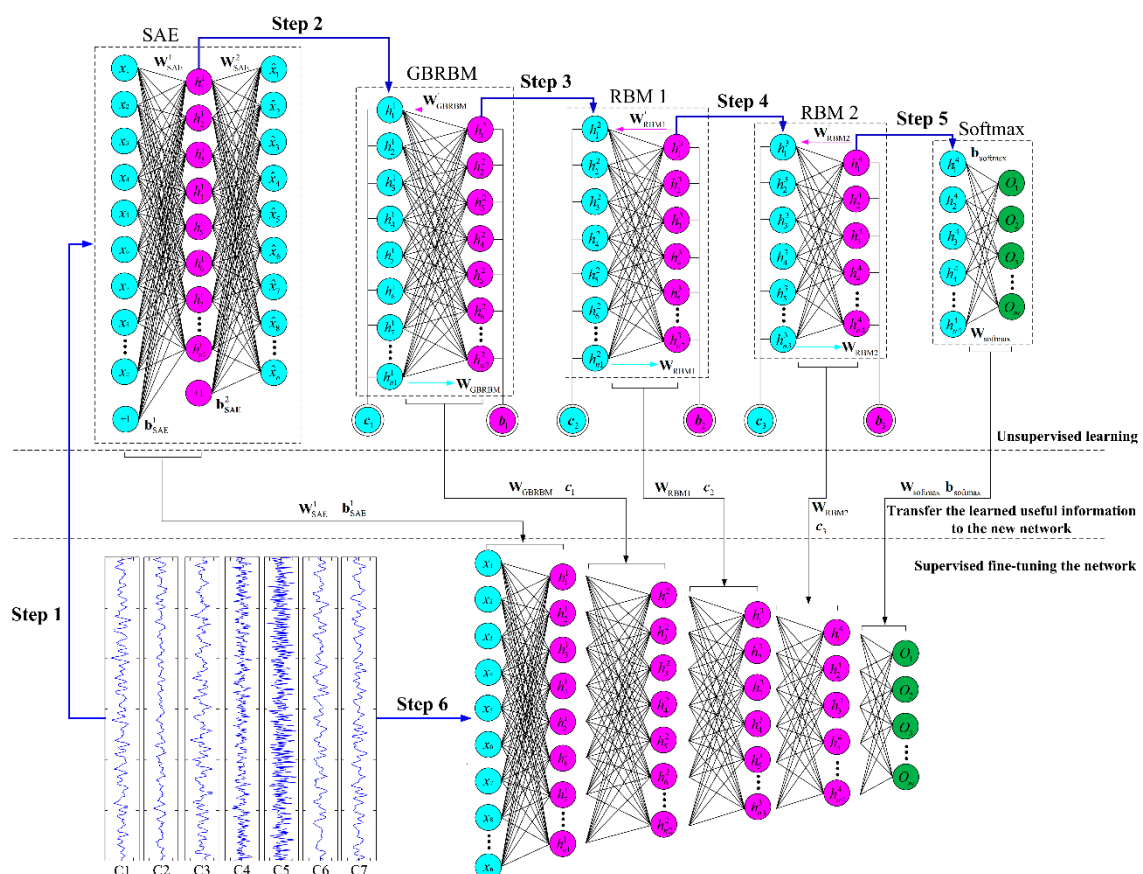


Figure 1. The framework of the proposed method.

As shown in Figure 1, the framework of the proposed method includes three parts: (1) unsupervised feature learning, (2) transfer the learned useful information to the new network, (3) supervised fine-tuning the restructured network. Stacked SAE, GBRBM and RBMs are combined to

work as a simultaneous signal processing and unsupervised feature extraction process. The blue circles in the figure represent the input layer neurons, the red circles represent the hidden layer neurons, and the green circles represent the output layer neurons. The entire diagnostic model has a total of 6 layers of neurons. The specific training process contains 6 steps as shown in Figure 1, raw vibration signals are first used for feature extraction through unsupervised learning, and the data is forwarded through the SAE, GBRBM, two-layer RBM and softmax layers, then fine-tune the weights and biases from unsupervised learning process of each layer according to the cross entropy error function.

The network training in Figure 1 consists of 6 steps. Table 1 shows the detailed calculation principle for the 6 steps, and also includes input values, output values, and parameters transferred for each layer. Figure 1 and Table 1 in combination gives a general understanding of the training procedure. First, the unsupervised learning is performed layer by layer. Then, the learned useful information is transferred into the new network. Finally, supervised fine-tune is performed to adjust the entire network. The detailed equations are shown in Table 1.

**Table 1.** Detailed process of proposed method.

<b>Overall process:</b>	
(1) Unsupervised: SAE→GBRBM→RBM <sup>(1,2)</sup> →Softmax→(2) Supervised: Back propagation	
<p><b>Step 1: SAE training</b>  <b>Input:</b> training data <math>\mathbf{x}</math>, <math>\mathbf{W}</math> and <math>\mathbf{b}</math>, <math>\lambda</math>, <math>\beta</math>, <math>\eta_1</math>, max-epochs<sup>(1)</sup>  <b>for</b> <math>i</math> to max-epochs<sup>(1)</sup></p> <ul style="list-style-type: none"> <li>• <math>\mathbf{h} = \text{sigm}(\mathbf{W}_1\mathbf{x} + \mathbf{b}_1)</math></li> <li>• <math>\hat{\mathbf{x}} = \text{sigm}(\mathbf{W}_2\mathbf{x} + \mathbf{b}_2)</math></li> <li>• <math>J_{\text{SAE}} = J_{\text{MSE}} + \lambda \cdot J_{\text{weight}} + \beta \cdot J_{\text{sparse}}</math></li> <li>• <math>\Delta w_{ij}^l = \Delta_1 \partial J_{\text{sparse}} / \partial w_{ij}^l</math>, <math>\Delta b_i^l = \Delta_1 \partial J_{\text{sparse}} / \partial b_i^l</math></li> </ul> <p><b>end</b>  <b>Output:</b> <math>\mathbf{h}_{\text{SAE}}</math>, <math>\mathbf{W}_1^{\text{SAE}}</math>, <math>\mathbf{b}_1^{\text{SAE}}</math></p> <p><b>Step 2: GBRBM training</b>  <b>Input:</b> <math>\mathbf{h}_{\text{SAE}}</math>, max-epochs<sup>(2)</sup>, <math>w_{ij}^1</math>, <math>c_i^1</math>, <math>b_j^1</math>, <math>\sigma_i^2</math>, <math>\eta_2</math>, <math>\alpha_1</math>  <b>for</b> <math>i</math> to max-epochs<sup>(2)</sup></p> <ul style="list-style-type: none"> <li>• <math>p(v_i = 1 \mathbf{h}) = N(v_i, c_i^1 + \sum_j w_{ij}^1 \cdot h_j, \sigma_i^2)</math></li> <li>• <math>p(h_j = 1 \mathbf{v}) = \text{sigm}(b_j^1 + \sum_i \frac{v_i}{\sigma_i^2} w_{ij}^1)</math></li> <li>• <math>\Delta w_{ij}^{\text{new}} = \alpha_1 \Delta w_{ij}^1 + \eta_2 (&lt; v_i^{(0)} h_j^{(0)} - v_i^{(1)} h_j^{(1)} &gt;)</math></li> <li>• <math>\Delta b_j^{\text{new}} = \alpha_1 \Delta b_j^1 + \eta_2 (&lt; h_j^{(0)} - h_j^{(1)} &gt;)</math></li> <li>• <math>\Delta c_i^{\text{new}} = \alpha_1 \Delta c_i^1 + \eta_2 (&lt; v_i^{(0)} - v_i^{(1)} &gt;)</math></li> </ul> <p><b>end</b>  <b>Output:</b> <math>\mathbf{h}_{\text{GBRBM}}</math>, <math>\mathbf{W}_{\text{GBRBM}}</math>, <math>\mathbf{b}_{\text{GBRBM}}</math></p> <p><b>Step 3: RBM1 training</b>  <b>Input:</b> <math>\mathbf{h}_{\text{GBRBM}}</math>, max-epochs<sup>(3)</sup>, <math>w_{ij}^2</math>, <math>c_i^2</math>, <math>b_j^2</math>, <math>\eta_3</math>, <math>\alpha_2</math>  <b>for</b> <math>i</math> to max-epochs<sup>(3)</sup></p> <ul style="list-style-type: none"> <li>• <math>p(v_i = 1 \mathbf{h}) = \text{sigm}(c_i^2 + \sum_j w_{ij}^2 h_j)</math></li> <li>• <math>p(h_j = 1 \mathbf{v}) = \text{sigm}(b_j^2 + \sum_i v_i w_{ij}^2)</math></li> <li>• The update process of <math>\Delta w_{ij}^{\text{new}}</math>, <math>\Delta b_j^{\text{new}}</math> and <math>\Delta c_i^{\text{new}}</math> is similar to GBRBM</li> </ul> <p><b>end</b>  <b>Output:</b> <math>\mathbf{h}_{\text{RBM1}}</math>, <math>\mathbf{W}_{\text{RBM1}}</math>, <math>\mathbf{b}_{\text{RBM1}}</math></p>	<p><b>Step 4: RBM2 training</b>  <b>Input:</b> <math>\mathbf{h}_{\text{RBM1}}</math>, max-epochs<sup>(4)</sup>, <math>w_{ij}^3</math>, <math>c_i^3</math>, <math>b_j^3</math>, <math>\eta_4</math>, <math>\alpha_3</math></p> <ul style="list-style-type: none"> <li>• The training process is similar to step 3.</li> </ul> <p><b>Output:</b> <math>\mathbf{h}_{\text{RBM2}}</math>, <math>\mathbf{W}_{\text{RBM2}}</math>, <math>\mathbf{b}_{\text{RBM2}}</math></p> <p><b>Step 5: Softmax layer</b>  <b>Input:</b> <math>\mathbf{h}_{\text{RBM2}}</math>, <math>w_{ij}^4</math>, <math>d_j</math></p> <ul style="list-style-type: none"> <li>• <math>y_j = \text{softmax}(\sum_{i=1}^p (h_i w_{ij}^4 + d_j))</math></li> <li>• <math>\text{softmax}(z_i) = (e^{z_i} / \sum_{j=1}^q e^{z_j})</math></li> </ul> <p><b>Output:</b> <math>\mathbf{y}</math>, <math>\mathbf{W}_{\text{softmax}}</math>, <math>\mathbf{d}</math></p> <p><b>Step 6: Back propagation</b>  <b>Input:</b> <math>\mathbf{y}</math>, max-epochs<sup>(5)</sup>, <math>\eta_5</math>  <b>for</b> <math>i</math> to max-epochs<sup>(5)</sup></p> <ul style="list-style-type: none"> <li>• <math>\mathbf{W}_1^{\text{SAE}}</math>, <math>\mathbf{b}_1^{\text{SAE}}</math>, <math>\mathbf{W}_{\text{GBRBM}}</math>, <math>\mathbf{b}_{\text{GBRBM}}</math>, <math>\mathbf{W}_{\text{RBM1}}</math>, <math>\mathbf{b}_{\text{RBM1}}</math>, <math>\mathbf{W}_{\text{RBM2}}</math>, <math>\mathbf{b}_{\text{RBM2}}</math>, <math>\mathbf{W}_{\text{softmax}}</math>, <math>\mathbf{d}</math> as the weight and bias of fully connected DNN.</li> <li>• <math>E_{\text{cross-entropy}} = -((o \log y) + (1 - o) \log (1 - y))</math></li> <li>• <math>\Delta w = \frac{\partial E_{\text{cross-entropy}}}{\partial w}</math>, <math>\Delta b = \frac{\partial E_{\text{cross-entropy}}}{\partial b}</math></li> </ul> <p><b>end all</b>  <b>Output: the trained network</b></p> <p><b>Step 7: Test the trained network with test sample</b></p>

## 2.2. Sparse Autoencoder

Sparse autoencoder (SAE) [33,34] is an unsupervised learning network mainly used for data dimensionality reduction and feature extraction. The SAE includes three layers: input layer ( $n + 1$  neuron), hidden layer ( $m + 1$  neuron,  $m < n$ ), and output layer ( $n$  neurons). Figure 1 show the structure of SAE, which can be seen to contain two processes of encoding and decoding.

The encoding process of SAE can be implemented by Equation (1), and the decoding process can be implemented by Equation (2).

$$\mathbf{h} = \text{sigm}(\mathbf{W}_1\mathbf{x} + \mathbf{b}_1) \quad (1)$$

$$\hat{\mathbf{x}} = \text{sigm}(\mathbf{W}_2\mathbf{h} + \mathbf{b}_2) \quad (2)$$

where  $\mathbf{x}$  is the input matrix,  $\mathbf{W}_1$  and  $\mathbf{b}_1$  are the weight matrix and bias vector between input layer and hidden layer,  $\mathbf{h}$  is the hidden matrix,  $\mathbf{W}_2$  and  $\mathbf{b}_2$  are the weight matrix and bias vector between hidden layer and output layer, and  $\hat{\mathbf{x}}$  is the output matrix; function  $\text{sigm}(\cdot) = 1/(1 + e^{-z})$ .

When the mean square error (MSE) is used as the loss function of SAE, the expected processing results usually cannot be achieved. In order to make SAE perform better, a new loss function is designed as Equation (3), which consists of three parts:  $J_{\text{MSE}}$ ,  $J_{\text{weight}}$ , and  $J_{\text{sparse}}$  [35]. The purpose of using  $J_{\text{weight}}$  is to control the value of the connected weights to avoid overfitting [36]. The added  $J_{\text{sparse}}$  is a sparsity penalty term, which can make SAE learn more features from the input by forcing SAE to maintain a degree of sparsity [37,38].

$$J_{\text{SAE}} = J_{\text{MSE}} + \lambda \cdot J_{\text{weight}} + \beta \cdot J_{\text{sparse}} \quad (3)$$

where  $J_{\text{MSE}}$  is the mean square error term as show in Equation (4),  $J_{\text{weight}}$  is the weight penalty item as show in Equation (5),  $J_{\text{sparse}}$  is the sparsity penalty term as show in Equation (6),  $\lambda$  is the regularization parameter of weight term, and  $\beta$  is the coefficient of sparsity penalty term.

$$J_{\text{MSE}} = \frac{1}{2s} \sum_{i=1}^s \|x_i - \hat{x}_i\| \quad (4)$$

$$J_{\text{weight}} = \frac{1}{2} \sum_{l=1}^{k-1} \sum_{j=1}^{n_l} \sum_{i=1}^{n_{l-1}} W_{ij}^l \quad (5)$$

$$J_{\text{sparse}} = \sum_{j=1}^m \text{KL}(\rho \parallel \hat{\rho}_j) = \sum_{j=1}^m \left( \rho \log \left( \frac{\rho}{\hat{\rho}_j} \right) + (1 - \rho) \log \left( \frac{1 - \rho}{1 - \hat{\rho}_j} \right) \right) \quad (6)$$

$$\hat{\rho}_j = \frac{1}{s} \sum_{i=1}^s h_{ij} \quad (7)$$

where  $s$  is the sample size of training set,  $k$  is the number of layers in the network,  $n_l$  is the neurons in layer  $l$ ,  $\rho$  is the set neuron sparsity parameter, and  $\hat{\rho}_j$  is the sparsity of the  $j$ -th neuron as show in Equation (7).

## 2.3. Develop the GBRBM based on RBM

Restricted Boltzmann Machine (RBM) is the basic component of the deep belief network (DBN) [39,40]. Similar to the SAE, it is also an unsupervised learning network that can be used for feature extraction. The RBM contains two layers: visible layer (contains  $n$  visible units) and hidden layer (contains  $m$  hidden units). The neurons in the same layer are not connected, and neurons in different layer are connected in each other. The weight matrix connecting the two layers is denoted by  $\mathbf{W}$ , the bias vector of the visible layer is denoted by  $\mathbf{c}$ , and the bias vector of the hidden layer is denoted by  $\mathbf{b}$ .

Inspired by statistical physics, it can be found that any probability distribution can be transformed into an energy-based model. The joint probability distribution of the visible layer and the hidden layer is proportional to the energy equation [41], as shown in Equation (8). And the joint probability distribution of  $\mathbf{v}$  and  $\mathbf{h}$  can be obtained as shown in Equation (9).

$$-\log P(\mathbf{v}, \mathbf{h}) \propto E(\mathbf{v}, \mathbf{h}|\theta) = -\sum_{i=1}^n c_i v_i - \sum_{j=1}^m b_j h_j - \sum_{i=1}^n \sum_{j=1}^m v_i w_{ij} h_j \quad (8)$$

$$P(\mathbf{v}, \mathbf{h}|\theta) = \frac{1}{Z(\theta)} \exp(-E(\mathbf{v}, \mathbf{h}|\theta)) \quad (9)$$

where  $v_i$  is the visible layer unit,  $h_j$  is the hidden layer unit,  $w_{ij}$  is the weights between visible layer and hidden layer,  $c_i$  and  $b_j$  are the bias of two layers;  $m$  hidden units in hidden layer,  $n$  visible units in visible layers,  $\theta = \{w_{ij}, c_i, b_j\}$  are the parameters of RBM, and  $Z(\theta) = \sum_n \sum_m \exp(-E(\mathbf{v}, \mathbf{h}|\theta))$  is a partition function.

The probability function of the visible layer is given by Equation (10).

$$\begin{aligned} P(\mathbf{v}; \theta) &= \sum_h P(\mathbf{v}, \mathbf{h}; \theta) \\ &= \frac{1}{Z(\theta)} \sum_h \exp\left(\sum_{i=1}^n c_i v_i + \sum_{j=1}^m b_j h_j + \sum_{i=1}^n \sum_{j=1}^m v_i w_{ij} h_j\right) \\ &= \frac{1}{Z(\theta)} \exp\left(\sum_{i=1}^n c_i v_i\right) \times \prod_{j=1}^m \sum_h \exp\left(b_j h_j + \sum_{i=1}^n v_i w_{ij} h_j\right) \end{aligned} \quad (10)$$

Combining Equations (9) and (10), the conditional probability of the hidden layer can be obtained as shown in Equation (11).

$$\begin{aligned} P(\mathbf{h}|\mathbf{v}; \theta) &= \frac{P(\mathbf{v}, \mathbf{h}; \theta)}{P(\mathbf{v}; \theta)} \\ &= \prod_j \frac{\exp\left(b_j h_j + \sum_{i=1}^n v_i w_{ij} h_j\right)}{\sum_h \left(b_j h_j + \sum_{i=1}^n v_i w_{ij} h_j\right)} = \prod_j P(h_j|v) \end{aligned} \quad (11)$$

Similarly, the conditional probability of the visible layer can be based on the joint probability of  $\mathbf{v}$  and  $\mathbf{h}$  divided by independent probability of hidden layer, as show in Equation (12).

$$P(\mathbf{v}|\mathbf{h}; \theta) = \frac{P(\mathbf{v}, \mathbf{h}; \theta)}{P(\mathbf{h}; \theta)} = \prod_i P(v_i|h) \quad (12)$$

The neurons in the same layer are not connected, meaning that the units are conditionally independent. So the conditional probability of the visible layer and hidden layer can be calculated by Equations (13) and (14).

$$P(v_i = 1|\mathbf{h}) = \text{sigm}\left(c_i + \sum_j w_{ij} h_j\right) \quad (13)$$

$$P(h_j = 1|\mathbf{v}) = \text{sigm}\left(b_j + \sum_i v_i w_{ij}\right) \quad (14)$$

where  $\text{sigm}(x) = 1/(1 + \exp(-x))$  is the sigmoid function.

The parameter update of the RBM can be obtained by performing a stochastic gradient descent on the negative log-likelihood probability of the training data. The gradient of the negative log probability visible layer to the network parameters can be calculated by Equations (15)–(17). The value of  $\langle \cdot \rangle_{\text{data}}$



is easy to get, but the value of  $\langle \cdot \rangle_{\text{model}}$  is difficult to get. Therefore, the contrastive divergence (CD) algorithm was proposed by Hinton [42].

$$\frac{\partial \log p(\mathbf{v}; \theta)}{\partial w_{ij}} = (\langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}}) \quad (15)$$

$$\frac{\partial \log p(\mathbf{v}; \theta)}{\partial \mathbf{b}} = (\langle h_j \rangle_{\text{data}} - \langle h_j \rangle_{\text{model}}) \quad (16)$$

$$\frac{\partial \log p(\mathbf{v}; \theta)}{\partial \mathbf{c}} = (\langle v_i \rangle_{\text{data}} - \langle v_i \rangle_{\text{model}}) \quad (17)$$

where  $\langle \cdot \rangle_{\text{data}}$  indicates expectations for data distribution and  $\langle \cdot \rangle_{\text{model}}$  is the expectation of the distribution of the model definition.

Both the visible layer and the hidden layer of RBM are binary layers. It is not appropriate to construct the RBM with the binary visible layer when the input is a continuous valued data. So this paper is to develop the Gauss-Binary RBM (GBRBM) [43–45] instead of standard RBM, and the energy function of the standard RBM in Equation (8) is changed to Equation (18).

$$E(\mathbf{v}, \mathbf{h} | \theta) = - \sum_{i=1}^n \frac{(v_i - c_i)^2}{2\sigma_i^2} - \sum_{j=1}^m b_j h_j - \sum_{i=1}^n \sum_{j=1}^m \frac{v_i}{\sigma_i^2} w_{ij} h_j \quad (18)$$

where  $\sigma_i^2$  is the variance of Gaussian distribution.

With the energy equation in Equation (18), the conditional probability between the visible layer and the hidden layer can be obtained according to the derivation process in Section 2.2.

$$P(v_i = v | \mathbf{h}) = N\left(v; c_i + \sum_j w_{ij} h_j, \sigma_i^2\right) \quad (19)$$

$$P(h_j = 1 | \mathbf{v}) = \text{sigm}\left(b_j + \sum_i \frac{v_i}{\sigma_i^2} w_{ij}\right) \quad (20)$$

where  $N(\cdot, \mu, \sigma_i^2)$  is Gaussian distribution, also called normal distribution,  $\mu$  is the mean, and  $\sigma_i^2$  is the variance.

The softmax classification layer is commonly used in the last layer of the neural network, and its working principle is shown in Equations (21) and (22).

$$y_j = \text{softmax}\left(\sum_{i=1}^p (h_i w_{ij} + d_j)\right) \quad (21)$$

$$\text{softmax}(z_i) = e^{z_i} / \sum_{j=1}^q e^{z_j} \quad (22)$$

where  $w_{ij}$  and  $d_j$  are weights and bias of softmax layer,  $h_i$  is the input of softmax layer,  $p$  is the number of neurons in input layer, and  $q$  is the number of neurons in output layer.

### 3. Experiment Setup and Data Acquisition

In this paper, vibration data collected from experiments of seven gears with early gear pitting faults on a gear test rig were used to validate the proposed method. Figure 2 shows the gear test rig and the seven gears with the early gear pitting faults. The gearbox in the test rig consists of a pair of spur gears. The pinion gear is the driving gear (including 40 teeth, module 3 mm), and the large gear is the driven gear (including 72 teeth, module 3 mm). The gearbox is powered by two Siemens servo motors with a power of 45 kW. Motor 1 is the driving motor and motor 2 is the loading motor.

The gearbox is equipped with a lubrication and cooling system. The tri-axial acceleration sensor was mounted on the gearbox housing (the red box in the figure) with a sampling rate of 10240 Hz, and the vibration signals in the three directions of X, Y and Z were collected.

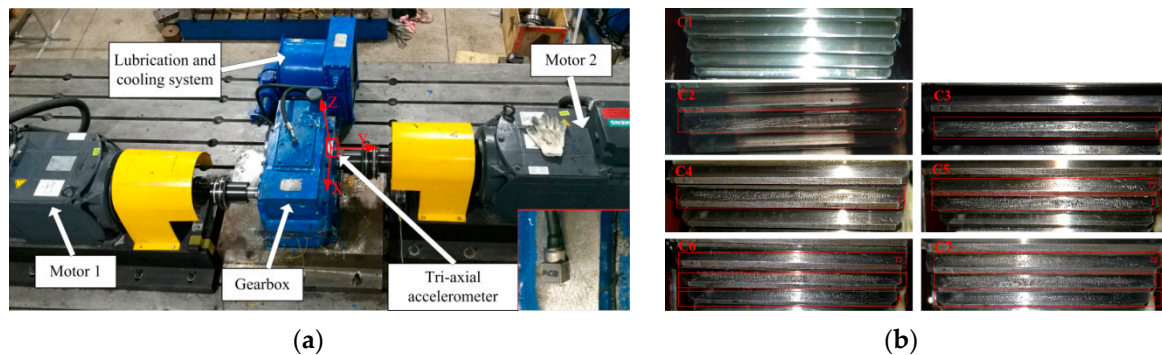


Figure 2. (a) Experimental test rig (b) gear pitting type.

The gear pitting faults were artificially manufactured by the drill on the driven gear surface. The specific conditions of the gear pitting faults are shown in Table 2. The fault degree is gradually increased and the latter one fault includes all of the previous fault conditions.

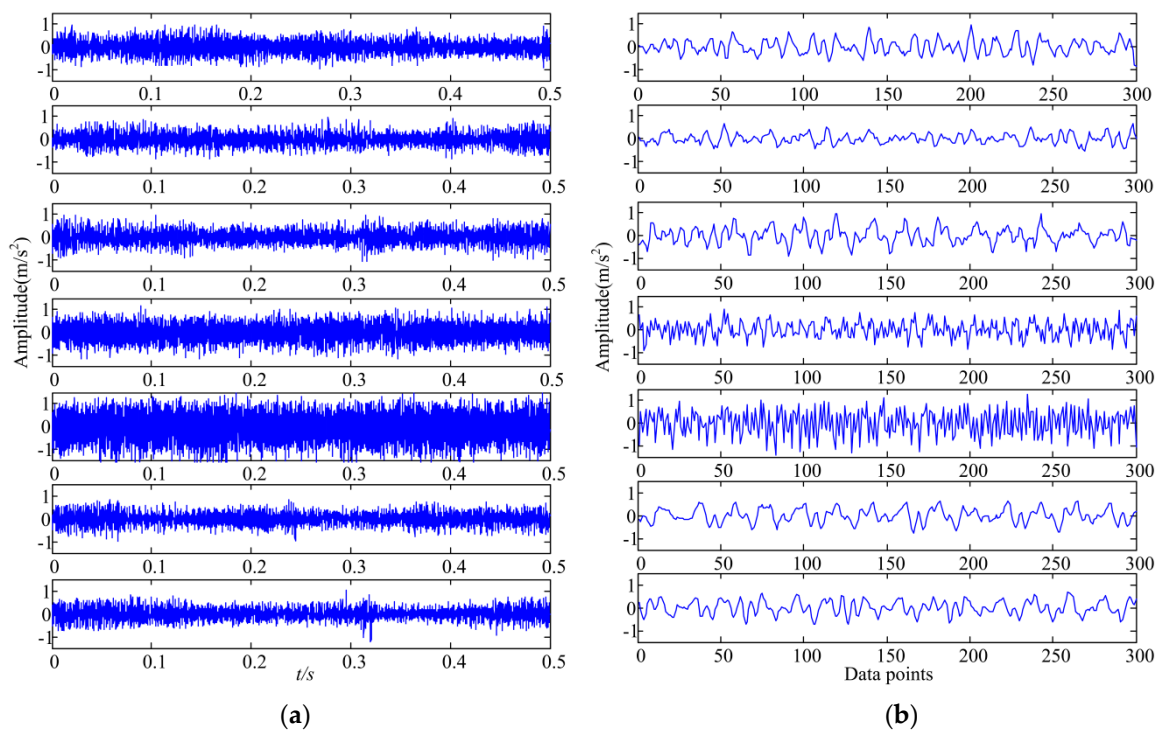
Table 2. Driven gear pitting type.

Label	Gear Pitting Type		
	72th Tooth	First Tooth	Second Tooth
C1	healthy	healthy	healthy
C2	healthy	10% in middle	healthy
C3	healthy	30% in middle	healthy
C4	healthy	50% in middle	healthy
C5	10% in middle	50% in middle	healthy
C6	10% in middle	50% in middle	10% in middle
C7	30% in middle	50% in middle	10% in middle

The vibration signals were collected under 25 working conditions. The 25 working conditions included combinations of five speeds (100–500 rpm) and five torque levels (100–500 Nm). Taking the working condition of 500 rpm–500 Nm as an example, each of seven gear types performed five independent data acquisitions and resulted in a total of 35 sets (120,000 data points per set) of data. 80% of all the data was used for training and the remaining data was used for testing. Hence, a training data matrix of  $120,000 \times 28$  and testing data matrix of  $120,000 \times 7$  were generated.

If the data matrix is directly used as the inputs, the network will be complex and the training will be slow. Therefore each data set was divided into several segmentations. For the sampling rate of 10240 Hz and a rotation speed of 500 RPM, approximately 1200 data points per gear rotation can be computed. In each segment, 300 data points (quarter of the collected data per gear rotation) were included [46]. In this case, the training data matrix dimension was  $300 \times 11200$  and test data matrix dimension was  $300 \times 2800$ . Figure 3a shows sample vibration signals of the seven gears in Z-axis under 500 rpm–500 Nm working condition and Figure 3b represents one segment of the corresponding sample vibration signals.





**Figure 3.** Sample vibration signals of seven gear types in Z-axis under 500 rpm–500 Nm: (a) signal with length of 0.5 s, (b) signal segment containing 300 data points.

## 4. Results and Discussion

### 4.1. PCA Data Visualization During the Training Process

To show the effectiveness by stacking SAE and GBRBM for extracting useful gear pitting fault information from the raw vibration signals, the network was trained with data from working condition 500 rpm–500 Nm. A total of six layers of neurons constitute the proposed diagnostic model, as shown in Figure 1. The structure of the proposed diagnostic model had the following structure: SAE:  $300 \times 300$  (300 neurons in the input layer and 300 neurons in the hidden layer), GRRBM:  $300 \times 200$  (300 neurons in the visible layer and 200 neurons in the hidden layer), RBM 1:  $200 \times 100$  (200 neurons in the visible layer and 100 neurons in the hidden layer), RBM 2:  $100 \times 50$  (100 neurons in the visible layer and 50 neurons in the hidden layer), Softmax:  $50 \times 7$  (50 neurons in the input layer and seven neurons in the output layer). The size of the weight matrix and the biases were determined by the structure of the proposed model. The initial weights ( $W_1$  and  $W_2$ ) of SAE layer were randomly generated between 0 and 1. The initial weights of the softmax layer were randomly generated between 0 and 0.5. The remaining initial weights and biases were set to 0. The proposed diagnostic model was trained layer by layer. Steps 1, 2, 3, 4, and 6 were trained in 300 epochs, respectively. The parameter  $\lambda$  of SAE layer was set to be 0.005,  $\beta$  set to 1.5, and  $\rho$  set to 0.1. The learning rate of GBRBM was set to 0.005, the learning rate of RBM set to 0.5, and the learning rate of the back propagation process set to 0.05. The minimum training error of the back propagation process was set to 0.05. The entire network was calculated on a mini-batch with the batch size set to 100. There are many related parameters affecting the performance of the diagnostic model. The key parameters such as learning rate, structure of the network, and training epochs that have a great impact on the diagnostic results will be discussed in Section 4.3 below.

The outputs of each layer in the network structure were obtained and these outputs were further processed by PCA. The first two principal components of the PCA results are used to draw a scatter plot in Figure 4 to show the changes of data. The effectiveness of each layer of the network can be judged by observing the changes in the data through each layer of the neural network. In the

experiment, the training and testing of the diagnostic model were performed using MATLAB 2014a software. The PCA results shown in Figure 4 were also obtained using the MATLAB codes. All the computational experiments were carried out on a PC with Windows 7 system and a CPU of Intel(R) Core i5-6500 @ 3.2GHz.

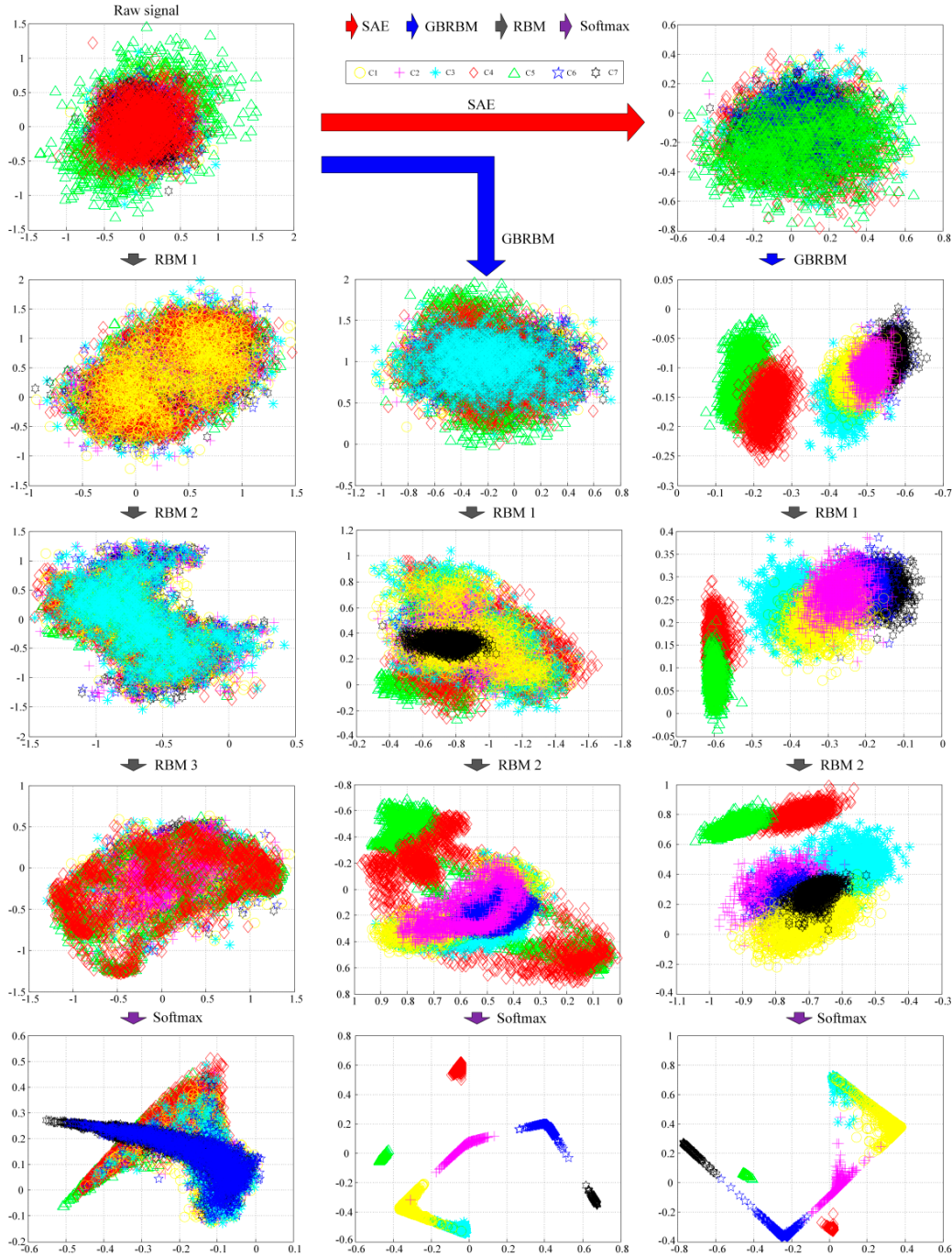
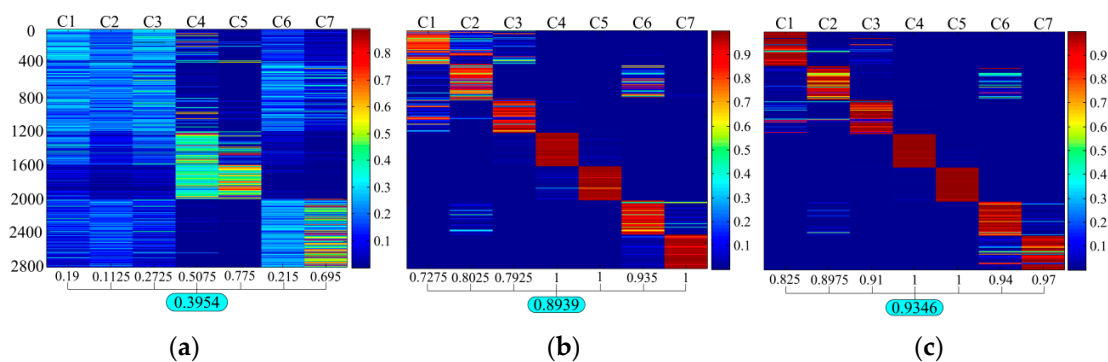


Figure 4. Each layer PCA result of three methods.

In Figure 4, three methods are shown. The first column in Figure 4 represents a standard DBN. The middle column represents the method with the first RBM layer of the standard DBN replaced with a GBRBM. The third column represents the proposed method by adding the SAE layer. As can be seen in Figure 4, the proposed method has the best fault separation result, and the separation result of the middle method is better than the standard DBN. Also seen from Figure 4, as the data moves from top

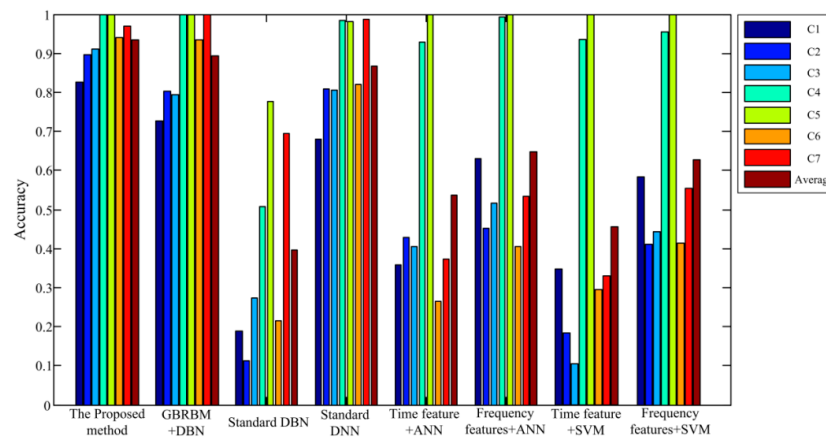
down, the level of the fault separation is getting better. Figure 5 shows the confusion matrix of the gear pitting fault diagnosis results of the three methods. Again, as shown in Figure 5, the proposed method has the best diagnosis accuracy of 0.9346, the method with the first RBM layer of the standard DBN replaced with a GBRBM has a diagnosis accuracy of 0.8939, and the standard DBN has the worst accuracy of 0.3954. Even though the confusion matrix shown in Figure 5b looks similar to that in Figure 5c obtained by the proposed method, the diagnostic accuracy for the confusion matrix shown in Figure 5b is 0.8939 while the diagnostic accuracy for the confusion matrix shown in Figure 5c is 0.9346. Therefore, the proposed method gives more accurate diagnosis results. As shown in Figure 5, the graph located at the 2nd row in the middle column represents the PCA result without going through the SAE layer, while the graph located at the 2nd row in the 3<sup>rd</sup> column represents the PCA result after being processed by the SAE layer. By comparing these two graphs in Figure 4, one should note that the PCA result obtained by the SAE layer in the proposed method gives a better pitting fault separation. The results have shown the effectiveness of SAE layer in the proposed method for extracting useful fault features when it is used for processing the vibration signals.



**Figure 5.** Confusion matrix of the fault diagnosis results: (a) standard DBN, (b) the first RBM layer of DBN replaced by GBRBM, (c) the proposed method.

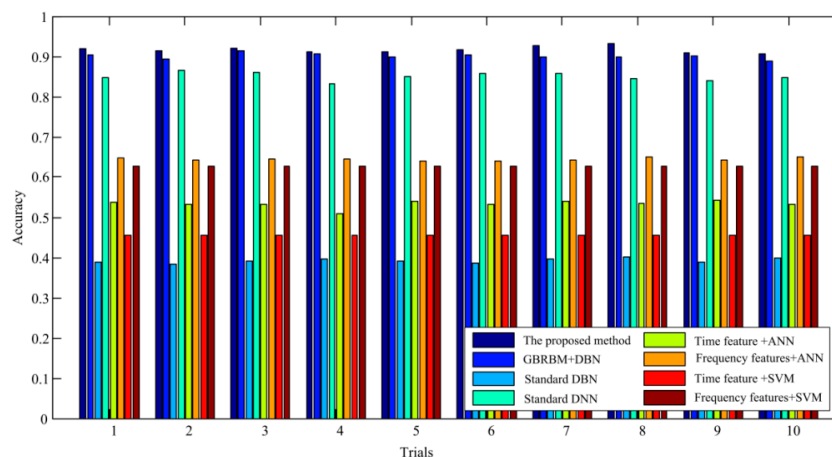
#### 4.2. Diagnostic Results of Proposed Method

Figure 6 shows the diagnostic accuracy of the proposed method and the other seven traditional methods. The 7 traditional methods include: (1) The first RBM layer of DBN replaced by GBRBM, (2) standard DBN, (3) standard DNN, (4) ANN with time domain vibration features, (5) ANN with frequency domain vibration features, (6) SVM with time domain vibration features, and (7) SVM with frequency domain vibration features. The results include the diagnostic accuracy for each gear pitting fault condition under 500 rpm–500 Nm working condition and the averaged accuracy over seven gear pitting fault conditions. From Figure 6, in comparison with other methods, the performance of the proposed method is significantly better than other methods. It can also be seen that the diagnostic accuracy for gear pitting conditions C4 and C5 is maintained at a high level in various methods, indicating that they are easier to diagnose than other fault conditions. This can be explained by observing the vibration signal in Figure 3b. It can be found that the vibration signal of C4 and C5 are clearly distinguished from the other gear pitting fault signals.



**Figure 6.** Diagnosis accuracy for all seven gear pitting conditions and the averaged accuracy under 500 rpm–500 Nm working condition.

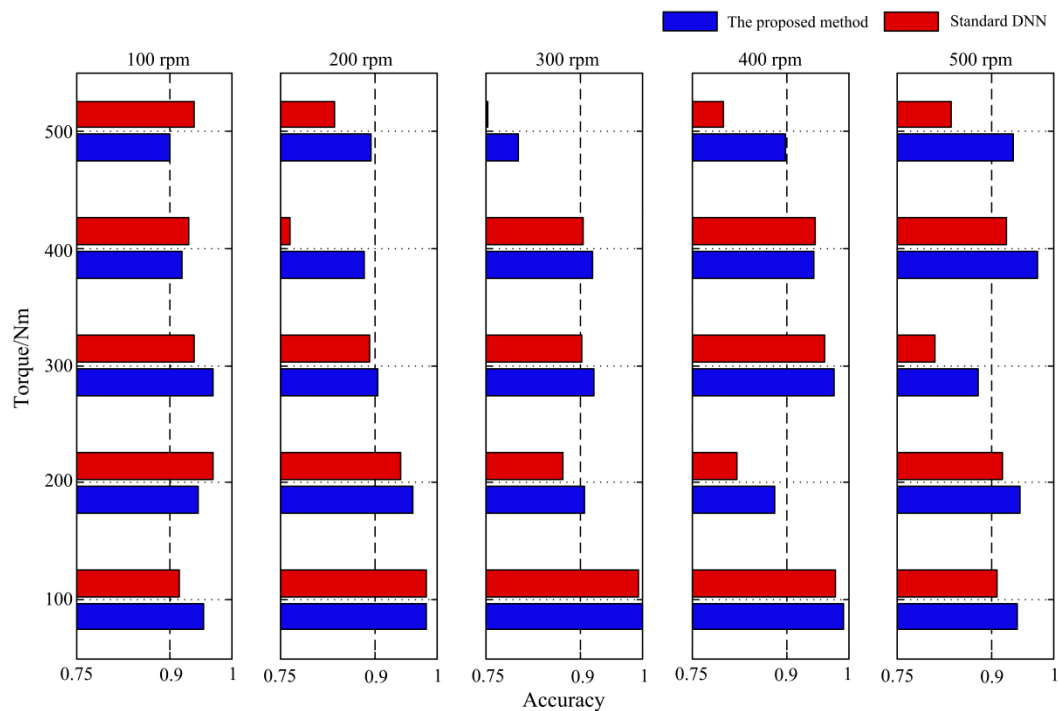
Figure 7 shows the averaged diagnostic accuracy over all seven gear pitting conditions under 500 rpm–500 Nm working condition in ten trials with eight different methods. It can be seen that the proposed method has the highest diagnostic accuracy. In comparison with the proposed method, the accuracy of the method with the first RBM layer of the standard DBN replaced with a GBRBM is slightly lower. The standard DNN methods also have more prominent diagnosis results.



**Figure 7.** Averaged diagnosis accuracy of the ten trials under 500 rpm–500 Nm working condition.

As shown in Figures 6 and 7, among the methods compared with the proposed method, standard DNN has shown a competitive performance under the 500 rpm–500 Nm working condition. To show the performance of the proposed method in comparison with DNN for all the working conditions, the vibration signals under 25 working conditions were used to compute the diagnostic accuracy for both the proposed method and the standard DNN. The results are provided in Figure 8, Tables 3 and 4.

In Figure 8, the averaged diagnosis accuracy over seven gear pitting conditions under 25 working conditions is provided for both the proposed method and the standard DNN. Further, the average accuracy over all five torque levels for each speed in Figure 8 is computed and provided in Table 3. The average accuracy over all five speeds for each torque level in Figure 8 is computed and provided in Table 4.



**Figure 8.** Diagnostic accuracy under 25 working condition.

**Table 3.** Averaged accuracy under 5 speeds.

Speed	Proposed Method	Standard DNN
100 rpm	<b>0.9374</b>	0.9372
200 rpm	<b>0.9245</b>	0.8824
300 rpm	<b>0.9091</b>	0.8831
400 rpm	<b>0.9372</b>	0.9003
500 rpm	<b>0.9344</b>	0.8791

**Table 4.** Averaged accuracy under 5 torques.

Torque	Proposed Method	Standard DNN
100 Nm	<b>0.9729</b>	0.9546
200 Nm	<b>0.9279</b>	0.9036
300 Nm	<b>0.9294</b>	0.8997
400 Nm	<b>0.9275</b>	0.8935
500 Nm	<b>0.8848</b>	0.8307

It can be seen from Figure 8, Tables 3 and 4 that the average diagnostic accuracy of the proposed method is higher than that of the standard DNN under various speeds and torque conditions.

It can be seen from Tables 3 and 4 that the diagnostic accuracy under 100 Nm working condition can reach to 0.9729. In order to prove the repeatability of the diagnosis results, five consecutive diagnoses were performed for five working conditions under 100 Nm. The diagnostic results are shown in Table 5. The averaged diagnostic accuracy of the five diagnosis results under 100Nm working condition is 0.9744, indicating that the proposed diagnostic method has high diagnostic reliability.



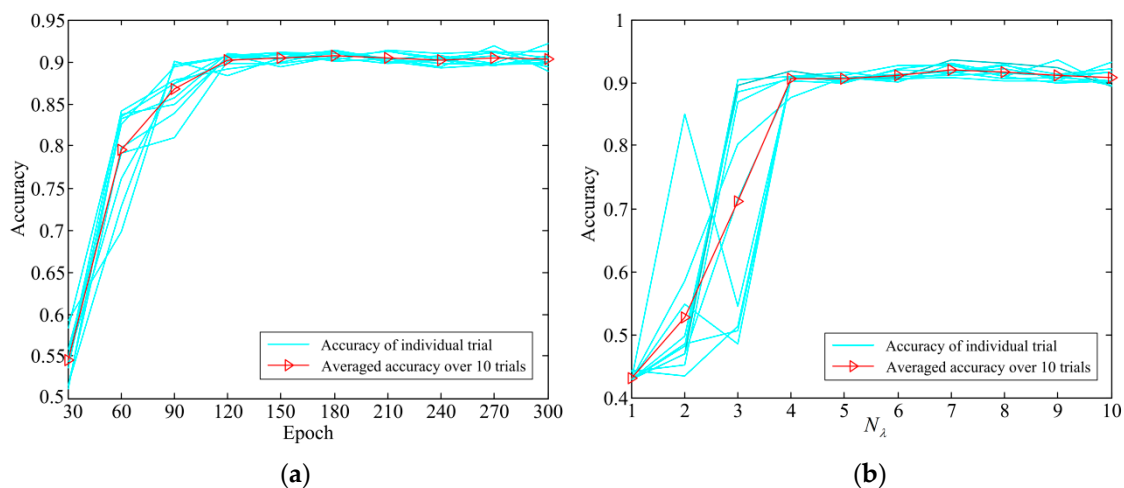
**Table 5.** Diagnosis accuracy of 5 trials under 5 working conditions.

Working Condition	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Row Average
100 rpm-100 Nm	0.9546	0.9554	0.9621	0.9354	0.9843	0.9584
200 rpm-100 Nm	0.9861	0.9636	0.9736	0.9236	0.9857	0.9665
100 Nm 300 rpm-100 Nm	1	0.9986	0.9975	0.9989	0.9961	0.9982
400 rpm-100 Nm	0.9954	0.9968	0.9950	0.9921	0.9961	0.9951
500 rpm-100 Nm	0.9582	0.9557	0.9446	0.9550	0.9557	0.9539
Column Average	0.9789	0.9740	0.9746	0.9610	0.9836	<b>0.9744</b>

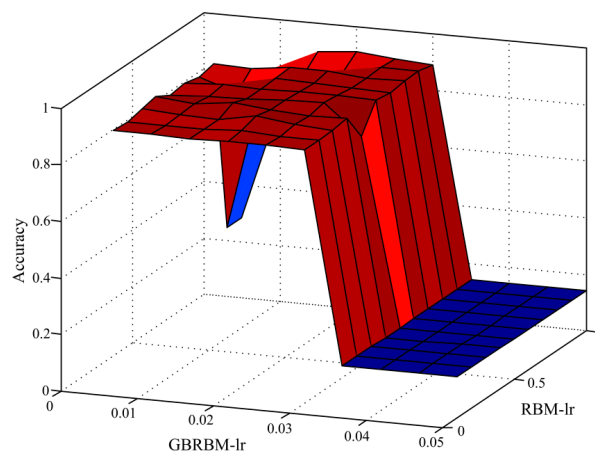
#### 4.3. The Effect of the Parameters on the Diagnostic Accuracy

To investigate effect of the parameters of the proposed method on the performance of the gear pitting fault diagnosis, experiments were performed. In the first experiment, diagnostic accuracy results with epochs increased from 30 to 300 in an increment of 5 were obtained. In the network structure of the proposed method, the number of neurons in the input layer and the output layer were 300 and 7.

In order to investigate the impact of the network structure on the performance of the proposed method, a structure parameter  $N_\lambda$  was designed to represent the middle layer. Let  $N_\lambda$  be an integer coefficient between 1 and 10. In this case, the network structure of the proposed method can be represented as:  $300-N_\lambda \times (30-20-10-5)-7$ . In the second experiment, diagnostic accuracy results with  $N_\lambda$  increased from 1 to 10 in an increment of 1 were obtained. The results of the first and second experiments are provided in Figure 9. From Figure 9a, the average accuracy of ten trials gradually increases when the training epochs increased from 30 to 120, and reached to constant level after 120 epochs. Figure 9b shows the effect of the parameter  $N_\lambda$  on the diagnostic accuracy of the network structure. When  $N_\lambda$  is increased from 1 to 4, the diagnostic accuracy is greatly improved. However, as  $N_\lambda$  reaches over 4, the improvement becomes insignificant.

**Figure 9.** Parameters affecting the diagnosis accuracy: (a) epochs, (b)  $N_\lambda$ .

To investigate the impact of the learning rate on the performance of the proposed method, in the third experiment, diagnostic accuracy results with the different learning rates (lr) in RBM and GBRBM were obtained. The results are provided in Figure 10. As seen from Figure 10, the learning rate of GBRBM has a greater impact on the diagnostic accuracy. When the learning rate of GBRBM is greater than 0.03, the accuracy decreased rapidly.



**Figure 10.** The influence of learning rate on the diagnosis accuracy.

## 5. Conclusions

In this paper, a novel method for early gear pitting fault diagnosis with raw vibration signals as direct inputs was presented. The method was developed by stacking a sparse autoencoder (SAE) and a Gauss-Binary restricted Boltzmann machine (GBRBM). The vibration data collected from the gear test rig was used to validate the diagnostic capability of the proposed method. The validation results have shown that the proposed method is capable of gear pitting fault diagnosis with high accuracy. The performance of the proposed method was also compared with other 7 methods including: (1) The first RBM layer of DBN replaced by GBRBM, (2) standard DBN, (3) standard DNN, (4) ANN with time domain vibration features, (5) ANN with frequency domain vibration features, (6) SVM with time domain vibration features, and (7) SVM with frequency domain vibration features. The results of the comparison have shown that the proposed method outperform the other methods in terms of the gear pitting fault diagnostic accuracy. The effect of parameters of the proposed method on the diagnostic performance of the proposed method was investigated and discussed in the paper.

**Author Contributions:** Conceptualization, D.H. and J.L.; methodology, J.L.; software, J.L.; validation, D.H., Y.Q., J.L. and X.L.; formal analysis, J.L.; investigation, D.H. and Y.Q.; resources, D.H. and Y.Q.; data curation, J.L. and X.L.; writing—original draft preparation, J.L.; writing—review and editing, D.H.; visualization, J.L.; supervision, D.H.; project administration, J.L. and X.L.; funding acquisition, D.H. and Y.Q.

**Funding:** This work was funded by the National Natural Science Foundation of China (No. 51675089 and No. 51505353).

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

## References

1. Khan, S.; Yairi, T. A review on the application of deep learning in system health management. *Mech. Syst. Signal Process.* **2018**, *107*, 241–265. [[CrossRef](#)]
2. Liu, R.; Yang, B.; Zio, E.; Chen, X. Artificial intelligence for fault diagnosis of rotating machinery: A review. *Mech. Syst. Signal Process.* **2018**, *108*, 33–47. [[CrossRef](#)]
3. Soualhi, A.; Medjaher, K.; Zerhouni, N. Bearing Health Monitoring Based on Hilbert-Huang Transform, Support Vector Machine, and Regression. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 52–62. [[CrossRef](#)]
4. Mejia-Barron, A.; Valtierra-Rodriguez, M.; Granados-Lieberman, D.; Olivares-Galvan, J.C.; Escarela-Perez, R. The application of EMD-based methods for diagnosis of winding faults in a transformer using transient and steady state currents. *Measurement* **2018**, *117*, 371–379. [[CrossRef](#)]
5. Bhattacharyya, A.; Pachori, R.; Upadhyay, A.; Acharya, U. Tunable-Q Wavelet Transform Based Multiscale Entropy Measure for Automated Classification of Epileptic EEG Signals. *Appl. Sci.* **2017**, *7*, 385. [[CrossRef](#)]
6. Gajjar, S.; Kulahci, M.; Palazoglu, A. Real-time fault detection and diagnosis using sparse principal component analysis. *J. Process Control* **2018**, *67*, 112–128. [[CrossRef](#)]

7. Bennacer, L.; Amirat, Y.; Chibani, A.; Mellouk, A.; Ciavaglia, L. Self-Diagnosis Technique for Virtual Private Networks Combining Bayesian Networks and Case-Based Reasoning. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 354–366. [[CrossRef](#)]
8. Denceux, T.; Kanjanatarakul, O.; Sriboonchitta, S. EK-NNclus: A clustering procedure based on the evidential K-nearest neighbor rule. *Knowl.-Based Syst.* **2015**, *88*, 57–69. [[CrossRef](#)]
9. Ziegler, J.; Gattringer, H.; Mueller, A. Classification of Gait Phases Based on Bilateral EMG Data Using Support Vector Machines. In *Proceedings of the 2018 7th IEEE International Conference on Biomedical Robotics and Biomechanics (Biorob)*; IEEE: Enschede, The Netherlands, 2018; pp. 978–983.
10. Ince, T.; Kiranyaz, S.; Eren, L.; Askar, M.; Gabbouj, M. Real-Time Motor Fault Detection by 1-D Convolutional Neural Networks. *IEEE Trans. Ind. Electron.* **2016**, *63*, 7067–7075. [[CrossRef](#)]
11. Zhang, R.; Peng, Z.; Wu, L.; Yao, B.; Guan, Y. Fault Diagnosis from Raw Sensor Data Using Deep Neural Networks Considering Temporal Coherence. *Sensors* **2017**, *17*, 549. [[CrossRef](#)]
12. Chen, Z.; Li, C.; Sanchez, R.-V. Gearbox Fault Identification and Classification with Convolutional Neural Networks. *Shock Vib.* **2015**, *2015*, 1–10. [[CrossRef](#)]
13. Chen, K.; Zhou, X.-C.; Fang, J.-Q.; Zheng, P.; Wang, J. Fault Feature Extraction and Diagnosis of Gearbox Based on EEMD and Deep Briefs Network. *Int. J. Rotating Mach.* **2017**, *2017*, 1–10. [[CrossRef](#)]
14. Wang, L.; Zhao, X.; Pei, J.; Tang, G. Transformer fault diagnosis using continuous sparse autoencoder. *SpringerPlus* **2016**, *5*. [[CrossRef](#)] [[PubMed](#)]
15. Tran, V.T.; AlThobiani, F.; Ball, A. An approach to fault diagnosis of reciprocating compressor valves using Teager–Kaiser energy operator and deep belief networks. *Expert Syst. Appl.* **2014**, *41*, 4113–4122. [[CrossRef](#)]
16. Han, D.; Zhao, N.; Shi, P. A new fault diagnosis method based on deep belief network and support vector machine with Teager–Kaiser energy operator for bearings. *Adv. Mech. Eng.* **2017**, *9*, 168781401774311. [[CrossRef](#)]
17. Shao, H.; Jiang, H.; Wang, F.; Wang, Y. Rolling bearing fault diagnosis using adaptive deep belief network with dual-tree complex wavelet packet. *ISA Trans.* **2017**, *69*, 187–201. [[CrossRef](#)] [[PubMed](#)]
18. Wang, S.; Xiang, J.; Zhong, Y.; Tang, H. A data indicator-based deep belief networks to detect multiple faults in axial piston pumps. *Mech. Syst. Signal Process.* **2018**, *112*, 154–170. [[CrossRef](#)]
19. Lee, D.; Lee, B.; Woo Shin, J. Fault Detection and Diagnosis with Modelica Language using Deep Belief Network. In *Proceedings of the 11th International Modelica Conference, Versailles, France, 21–23 September 2015*; pp. 615–623.
20. Ahmed, H.O.A.; Dennis Wong, M.L.; Nandi, A.K. Effects of deep neural network parameters on classification of bearing faults. In *Proceedings of the IECON—42nd Annual Conference of the IEEE Industrial Electronics Society, Florence, Italy, 23–26 October 2016*; pp. 6329–6334.
21. Tao, J.; Liu, Y.; Yang, D. Bearing Fault Diagnosis Based on Deep Belief Network and Multisensor Information Fusion. *Shock Vib.* **2016**, *2016*, 1–9. [[CrossRef](#)]
22. He, J.; Yang, S.; Gan, C. Unsupervised Fault Diagnosis of a Gear Transmission Chain Using a Deep Belief Network. *Sensors* **2017**, *17*, 1564. [[CrossRef](#)]
23. Chen, Z.; Deng, S.; Chen, X.; Li, C.; Sanchez, R.-V.; Qin, H. Deep neural networks-based rolling bearing fault diagnosis. *Microelectron. Reliab.* **2017**, *75*, 327–333. [[CrossRef](#)]
24. Deutsch, J.; He, M.; He, D. Remaining Useful Life Prediction of Hybrid Ceramic Bearings Using an Integrated Deep Learning and Particle Filter Approach. *Appl. Sci.* **2017**, *7*, 649. [[CrossRef](#)]
25. Geng, Z.; Li, Z.; Han, Y. A new deep belief network based on RBM with glial chains. *Inf. Sci.* **2018**, *463–464*, 294–306. [[CrossRef](#)]
26. Ren, H.; Chai, Y.; Qu, J.; Ye, X.; Tang, Q. A novel adaptive fault detection methodology for complex system using deep belief networks and multiple models: A case study on cryogenic propellant loading system. *Neurocomputing* **2018**, *275*, 2111–2125. [[CrossRef](#)]
27. Shao, H.; Jiang, H.; Zhang, H.; Duan, W.; Liang, T.; Wu, S. Rolling bearing fault feature learning using improved convolutional deep belief network with compressed sensing. *Mech. Syst. Signal Process.* **2018**, *100*, 743–765. [[CrossRef](#)]
28. Jiang, H.; Shao, H.; Chen, X.; Huang, J. A feature fusion deep belief network method for intelligent fault diagnosis of rotating machinery. *J. Intell. Fuzzy Syst.* **2018**, *34*, 3513–3521. [[CrossRef](#)]
29. Shao, H.; Jiang, H.; Lin, Y.; Li, X. A novel method for intelligent fault diagnosis of rolling bearings using ensemble deep auto-encoders. *Mech. Syst. Signal Process.* **2018**, *102*, 278–297. [[CrossRef](#)]

30. Maurya, S.; Singh, V.; Dixit, S.; Verma, N.K.; Salour, A.; Liu, J. Fusion of Low-level Features with Stacked Autoencoder for Condition based Monitoring of Machines. In Proceedings of the 2018 IEEE International Conference on Prognostics and Health Management (ICPHM), Seattle, WA, USA, 11–13 June 2018; pp. 1–8.
31. Shao, H.; Jiang, H.; Zhao, H.; Wang, F. A novel deep autoencoder feature learning method for rotating machinery fault diagnosis. *Mech. Syst. Signal Process.* **2017**, *95*, 187–204. [[CrossRef](#)]
32. Meng, Z.; Zhan, X.; Li, J.; Pan, Z. An enhancement denoising autoencoder for rolling bearing fault diagnosis. *Measurement* **2018**, *130*, 448–454. [[CrossRef](#)]
33. Sohaib, M.; Kim, J.-M. Reliable Fault Diagnosis of Rotary Machine Bearings Using a Stacked Sparse Autoencoder-Based Deep Neural Network. *Shock Vib.* **2018**, *2018*, 1–11. [[CrossRef](#)]
34. Saufi, S.R.; bin Ahmad, Z.A.; Leong, M.S.; Lim, M.H. Differential evolution optimization for resilient stacked sparse autoencoder and its applications on bearing fault diagnosis. *Meas. Sci. Technol.* **2018**, *29*, 125002. [[CrossRef](#)]
35. Gao, X.; Wang, H.; Gao, H.; Wang, X.; Xu, Z. Fault diagnosis of batch process based on denoising sparse auto encoder. In Proceedings of the 2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC), Nanjing, China, 18–20 May 2018; pp. 764–769.
36. Mahdi, M.; Genc, V.M.I. Post-fault prediction of transient instabilities using stacked sparse autoencoder. *Electr. Power Syst. Res.* **2018**, *164*, 243–252. [[CrossRef](#)]
37. Xu, L.; Cao, M.; Song, B.; Zhang, J.; Liu, Y.; Alsaadi, F.E. Open-circuit fault diagnosis of power rectifier using sparse autoencoder based deep neural network. *Neurocomputing* **2018**, *311*, 1–10. [[CrossRef](#)]
38. Amini, S.; Ghaernmaghami, S. Sparse Autoencoders Using Non-smooth Regularization. In Proceedings of the 2018 26th European Signal Processing Conference (EUSIPCO), Rome, Italy, 3–7 September 2018; pp. 2000–2004.
39. Shao, H.; Jiang, H.; Zhang, X.; Niu, M. Rolling bearing fault diagnosis using an optimization deep belief network. *Meas. Sci. Technol.* **2015**, *26*, 115002. [[CrossRef](#)]
40. Jiang, H.; Shao, H.; Chen, X.; Huang, J. Aircraft Fault Diagnosis Based on Deep Belief Network. In Proceedings of the 2017 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC), Shanghai, China, 16–18 August 2017; pp. 123–127.
41. Qin, X.; Zhang, Y.; Mei, W.; Dong, G.; Gao, J.; Wang, P.; Deng, J.; Pan, H. A cable fault recognition method based on a deep belief network. *Comput. Electr. Eng.* **2018**, *71*, 452–464. [[CrossRef](#)]
42. Hinton, G.E.; Osindero, S.; Teh, Y.-W. A Fast Learning Algorithm for Deep Belief Nets. *Neural Comput.* **2006**, *18*, 1527–1554. [[CrossRef](#)] [[PubMed](#)]
43. Li, Z.; Cai, X.; Liu, Y.; Zhu, B. A Novel Gaussian–Bernoulli Based Convolutional Deep Belief Networks for Image Feature Extraction. *Neural Process. Lett.* **2018**. [[CrossRef](#)]
44. Cho, K.H.; Raiko, T.; Ilin, A. Gaussian-Bernoulli deep Boltzmann machine. In Proceedings of the 2013 International Joint Conference on Neural Networks (IJCNN), Dallas, TX, USA, 4–9 August 2013; pp. 1–7.
45. Keronen, S.; Cho, K.; Raiko, T.; Ilin, A.; Palomaki, K. Gaussian-Bernoulli restricted Boltzmann machines and automatic feature extraction for noise robust missing data mask estimation. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 6729–6733.
46. Lu, W.; Wang, X.; Yang, C.; Zhang, T. A novel feature extraction method using deep neural network for rolling bearing fault diagnosis. In Proceedings of the 27th Chinese Control and Decision Conference (2015 CCDC), Qingdao, China, 23–25 May 2015; pp. 2427–2431.

