

Article

Compound Fault Diagnosis of Planetary Gearbox Based on Improved LTSS-BoW Model and Capsule Network

Guoyan Li ^{1,2,*}, Liyu He ¹, Yulin Ren ¹, Xiong Li ¹, Jingbin Zhang ¹ and Runjun Liu ¹

¹ Key Laboratory of Advance Transducers and Intelligent Control System, Ministry of Education, Taiyuan University of Technology, Taiyuan 030024, China; 15536645231@163.com (L.H.); renyulin991003@163.com (Y.R.); 19873375060@163.com (X.L.); zhangjingbin04@163.com (J.Z.); 13277234451@163.com (R.L.)

² School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

* Correspondence: liguoyan@tyut.edu.cn

Abstract: The identification of compound fault components of a planetary gearbox is especially important for keeping the mechanical equipment working safely. However, the recognition performance of existing deep learning-based methods is limited by insufficient compound fault samples and single label classification principles. To solve the issue, a capsule neural network with an improved feature extractor, named LTSS-BoW-CapsNet, is proposed for the intelligent recognition of compound fault components. Firstly, a feature extractor is constructed to extract fault feature vectors from raw signals, which is based on local temporal self-similarity coupled with bag-of-words models (LTSS-BoW). Then, a multi-label classifier based on a capsule network (CapsNet) is designed, in which the dynamic routing algorithm and average threshold are adopted. The effectiveness of the proposed LTSS-BoW-CapsNet method is validated by processing three compound fault diagnosis tasks. The experimental results demonstrate that our method can via decoupling effectively identify the multi-fault components of different compound fault patterns. The testing accuracy is more than 97%, which is better than the other four traditional classification models.

Keywords: planetary gearbox; compound fault diagnosis; capsule network; local temporal self-similarity; bag of words model



Citation: Li, G.; He, L.; Ren, Y.; Li, X.; Zhang, J.; Liu, R. Compound Fault Diagnosis of Planetary Gearbox Based on Improved LTSS-BoW Model and Capsule Network. *Sensors* **2024**, *24*, 940. <https://doi.org/10.3390/s24030940>

Academic Editors: Kim Phuc Tran, Athanasios Rakitzis and Khanh T. P. Nguyen

Received: 27 December 2023

Revised: 28 January 2024

Accepted: 30 January 2024

Published: 31 January 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Literature Review

Planetary gearboxes play an important role in mechanical equipment such as wind turbine, helicopter and construction machinery, which generally work under time-varying load conditions. The key parts of a planetary gearbox are prone to multiple structural damages such as wear, broke, pitting and crack, etc. due to the influence of long-term alternating stresses. The service performance of a planetary gearbox further endangers the operation safety of the entire mechanical equipment. Therefore, it is significant to diagnose the potential faults of a planetary gearbox [1,2].

The internal components of a planetary gearbox are varied and generally work together with a complex coupling relationship. The fault characteristics are coupled in that the failure of several components may simultaneously occur to different degrees. Moreover, the fault features could be impacted by multi-source excitations such as random impacts, time-varying load, strong noise, multi-interface attenuation and so on. As a result, it is very difficult to identify the compound fault of a planetary gearbox [3,4].

A series of studies have been carried out for the signal processing methods of a compound fault diagnosis. Since the fault signal is highly unstable with complex frequency components, time-frequency methods such as the wavelet transform, ensemble empirical mode decomposition (EEMD), symplectic geometric mode decomposition (SGMD), local

mean decomposition (LMD), local characteristic-scale decomposition (LCD), variational mode decomposition (VMD), etc. are mainly used. Teng et al. [5] proposed a modulation model based on wavelet transform, which provided an effective tool for wind power gearbox compound fault diagnosis; Zhao et al. [6] used the EEMD and feature fusion methods to diagnose the composite fault of rolling bearing; Pan et al. [7] proposed an SGMD signal decomposition algorithm to decompose the compound fault signals of rotating machinery; Huang et al. [8] combined recursive least squares (RLS) with LMD to diagnose the early fault of bearings; Wang et al. [9] proposed an improved LCD method to extract the early fault characteristics of bearings; Zhang et al. [10] combined VMD with adaptive maximum correlated kurtosis deconvolution (AMCKD) to detect the wind turbine rolling bearing faults. Above all, these methods mainly focus on the improvement of the signal decomposition ability. However, the subsequent compound fault separation and identification rely heavily on expert experience and knowledge, resulting in low recognition accuracy.

Deep learning (DL) has been used increasingly in the intelligent diagnosis of mechanical equipment. The typical DL-based methods for the intelligent diagnosis of a planetary gearbox include: Deep Belief Network (DBN) [11,12], Generative Adversarial Network (GAN) [13], Convolutional Neural Network (CNN) [14–16], Long Short-Term Memory (LSTM) [17], etc. In terms of compound fault diagnosis, Shao et al. [12] combined adaptive DBN and CNN to diagnose the multiple faults of rolling bearings; Zhao et al. [13] proposed a GAN model to improve the diagnosis performance under data imbalance conditions; Zhang et al. [14] combined fast spectral kurtosis (FSK) with multi-branch CNN for multi-fault diagnosis of wind turbine gearboxes.

However, most classification models treat compound fault as a new fault class and output single label, which cannot provide a true sense of decoupling identification of the compound fault. In fact, the compound fault is not exactly a new fault class since its fault information consists of corresponding fault characteristics of single faults. In addition, the training process of a DL-based model requires a large number of training samples. However, the fault samples are relatively rare in practice, and random combinations of different single faults can generate various compound faults. It is impractical to collect sufficient compound fault samples for classification model training. Therefore, it is necessary to propose a new intelligent diagnosis method, which is especially suitable for compound fault diagnosis, and has the following functions: (1) only the single fault samples are required for model training, and the trained model can use the fault knowledge learned from the labeled single fault samples to identify the fault components of compound fault test samples; (2) the model can predict multi-labels for test samples when making classification decisions.

The typical multi-label classification methods include binary relevance, multi-label K-NN, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Transformer structures [18,19], etc. Capsule Network (CapsNet) is a novel type of network proposed by Hinton et al. [20] in 2017. It utilizes capsule vectors rather than scalar neurons as the input and output of the network layers, which overcomes the problem that traditional networks cannot extract the spatial feature information. Meanwhile, it cancels the pooling layer to avoid the loss of valuable information, and can conduct multi-label outputs. It has been increasingly used in the fields of electroencephalography (EEG) emotion recognition [21], image and text classification [22,23], etc. In particular, it can identify and separate the overlapped objects [22], which is an important feature for the identification of compound faults.

In terms of single fault diagnosis, Liu et al. [24] proposed an improved multi-scale residual generative adversarial network (GAN) and feature enhancement-driven capsule network to solve the imbalanced fault data problem. Li et al. [25] proposed a dual convolutional–CapsNet for the fault diagnosis of a planetary gearbox under different rotation conditions. In terms of compound fault diagnosis, Liang et al. [26] integrated CapsNet with stockwell transform (ST) and data augmentation generative adversarial networks (DAGANs) to diagnose the single and simultaneous faults for a wind turbine gearbox. Xu et al. [27] developed an improved deep convolutional–CapsNet to diagnose the sun

gear–planet compound faults of an RV gearbox; Huang et al. [28] adopted a convolutional–CapsNet model with a multi-label classifier to decouple the gear-bearing compound faults of automotive transmission. Then, Huang et al. [29] combined deep CapsNet and ensemble learning to improve the compound fault identification accuracy of an automotive gearbox.

To achieve accurate identification, sufficient feature information needs to be fed into the primary layer to ensure the CapsNet is working efficiently, which depends on the feature extractor of the model. However, most models use the convolutional network to extract features from raw signals; the feature extraction process is considered as a “black box”, which has the limitation in compound fault feature extraction. As a result, the classification performance of the diagnosis model has been limited. This motivates us to develop a more suitable feature extractor to directly obtain high representation feature vectors from raw signals and ultimately increase the classification ability.

To address this issue, a new feature extractor is constructed to optimize CapsNet, which combines local temporal self-similarity (LTSS) and bag-of-words (BoW) methods, named LTSS-BoW, for feature extraction. This model is improved from the temporal self-similarity method [30], which has been successfully used to recognize the image action sequences due to its advantage of cross-view structural stability. In order to reduce the feature dimension and increase the computing efficiency, a sliding window is utilized to divide the raw time-series into a local subseries. On this basis, LTSS matrices of the subseries are constructed and the gradients of LTSS matrices are calculated. Then, the multi-dimensional LTSS feature vectors are obtained by moving the sliding window with a fixed step size to traverse the entire sample signal. The LTSS feature extraction leads to much data redundancy, and thus brings a large computation burden. Therefore, BoW is utilized to further improve computing efficiency, which has the advantages of strong anti-noise ability and good robustness [31,32]. Finally, the histogram feature vectors are treated as the inputs of the CapsNet layers.

1.2. Main Contributions of This Paper

The main contributions of this work are summarized as follows:

- (1) A novel framework, named LTSS-BoW-CapsNet, is proposed to intelligently identify the fault components contained in the compound fault signals of the planetary gearbox.
- (2) An LTSS-BoW-based feature extraction method is presented to increase the identification performance, which can be used to directly obtain high representation feature vectors from raw signals.
- (3) A multi-label classifier based on CapsNet is designed to predict multi-labels for compound fault classification decisions, in which the dynamic routing algorithm and average threshold are adopted.
- (4) Verification experiments are conducted to demonstrate the advantages of the proposed method.

1.3. Structure of the Rest of This Paper

The rest of this paper is organized as follows: Section 2 introduces the basic theoretical background of the LTSS-BoW feature extractor and CapsNet, and presents the overall diagnostic scheme of the proposed LTSS-BoW-CapsNet method in detail. Section 3 shows the experimental verification and comparative analysis results. The conclusions are drawn in Section 4.

2. Research Methodology

2.1. Compound Fault Description

Based on the author’s previous research works [33,34], Figure 1 exhibits the simulated vibration signals of a planetary gear set with the planet gear crack, sun gear crack and compound gear cracks, respectively. For the single crack fault cases, a series of abnormal impulses with a fixed period appear in the time-domain as the cracked tooth engages with the matching gear. As shown in Figure 1d, the compound fault-related features contain

the information of two kinds of single fault-induced features. However, it is not simply the superposition of single fault-induced features. When two cracked teeth are engaged simultaneously, the two types of anomalous pulses will be coupled and form new fault features; meanwhile, the single fault-induced features are also deformed. Therefore, it is difficult to identify the fault components contained in the compound fault signal.

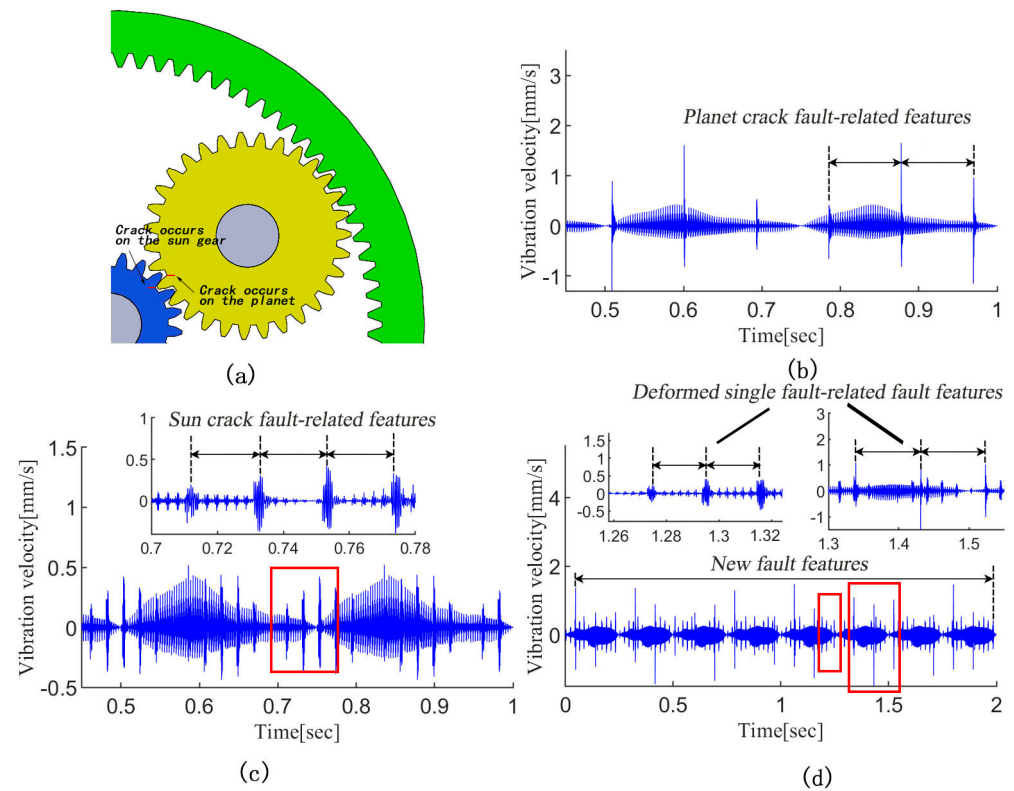


Figure 1. The simulated vibration signals of a planetary gear set with gear cracks. (a) Planetary gear set; (b) planet gear crack; (c) sun gear crack; (d) compound gear cracks.

The main limitations of existing compound fault diagnosis models are: (1) most researchers label compound faults as a new fault class, and the compound fault samples need to be fed into the network with other single fault samples for model training [12–14]. Therefore, the proposed network cannot work effectively if the compound fault training samples are insufficient. However, compound faults are not exactly a new fault class since it contains the fault information of single fault components. Hence, it is possible to use the fault knowledge learned from the labeled single fault samples to identify the fault components of the compound fault; (2) the traditional neural networks generally use a softmax classifier, which only identifies the most obvious fault class [26,27]. Therefore, it cannot output multiple independent labels at the same time, so it is unable to identify via decoupling the fault components contained inside the compound fault. In addition, it is more difficult to distinguish the fault components in the condition that the single fault-related features are close to each other.

2.2. Overall Framework of the Proposed Method

In order to intelligently identify the fault components contained in the compound fault of the planetary gearbox, this work proposed a novel diagnostic framework, which mainly includes two parts: the LTSS-BoW model is the feature extractor, and CapsNet makes multi-label classification decisions. The overall framework of the proposed method is shown in Figure 2. Each part is described in the following.

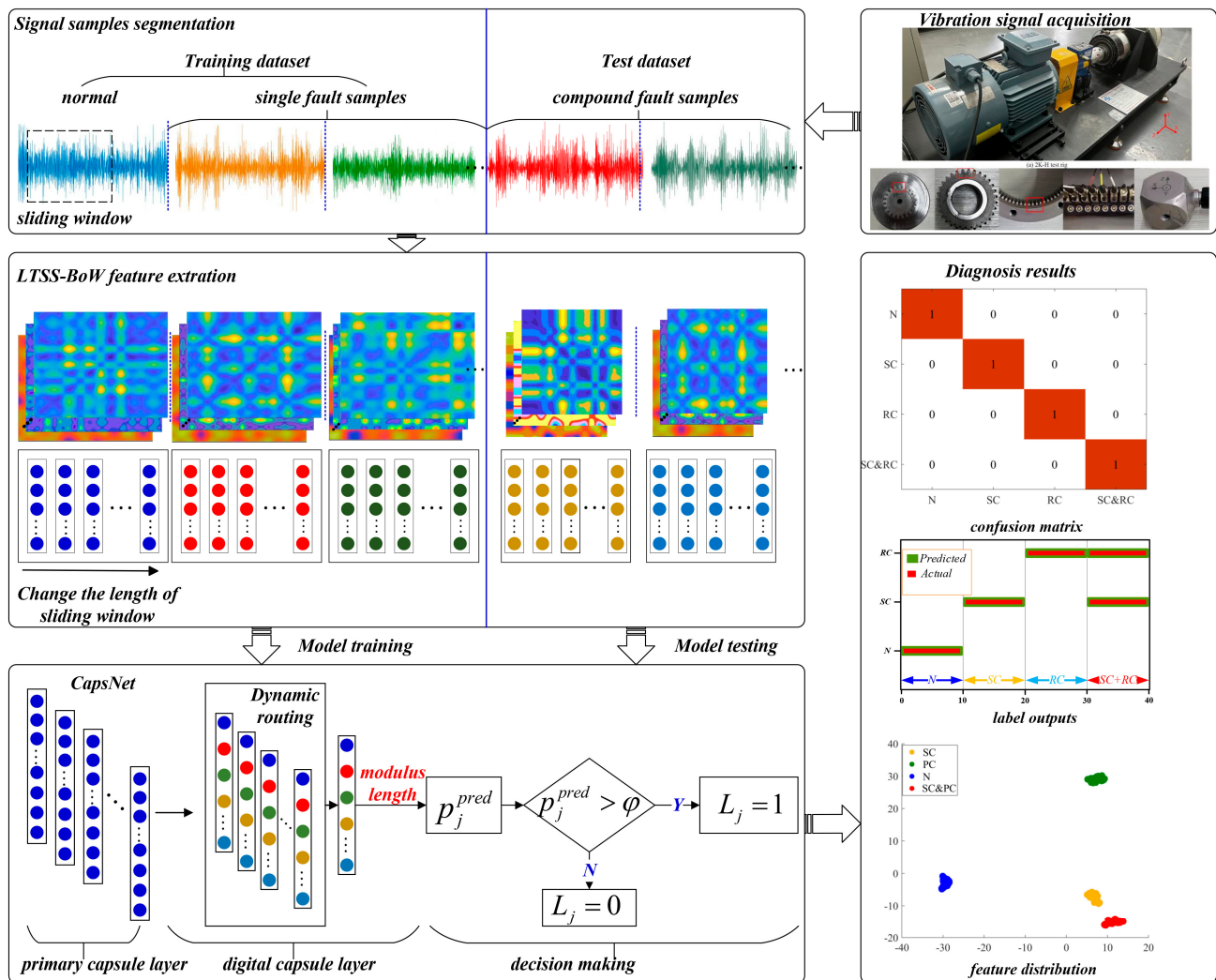


Figure 2. Overall framework of the proposed method.

2.3. LTSS-BoW Feature Extractor

2.3.1. LTSS Feature Extraction

LTSS is an optimized data representation method that utilizes local structural information of time-series. Compared with conventional statistical features, LTSS feature matrices contain more useful information including sequential characteristics and change trends.

Figure 3 shows the proposed LTSS feature extraction scheme, which includes three steps: (1) Construct the LTSS matrix from the raw signal. At first, the sliding window is utilized to divide the raw signal sample into a local subseries. On this basis, LTSS matrices of the subseries are constructed. (2) Extract the gradient feature of the LTSS matrix. The upper right triangular elements of the LTSS matrix are divided into several blocks. Then, the gradient of each block is calculated to construct the block-based descriptor. (3) Transform the signal sample to a sequence of LTSS feature vectors. The signal samples are represented as a sequence of LTSS feature vectors, and all these samples are then gathered together to form a feature space. The detailed steps are given as follows.

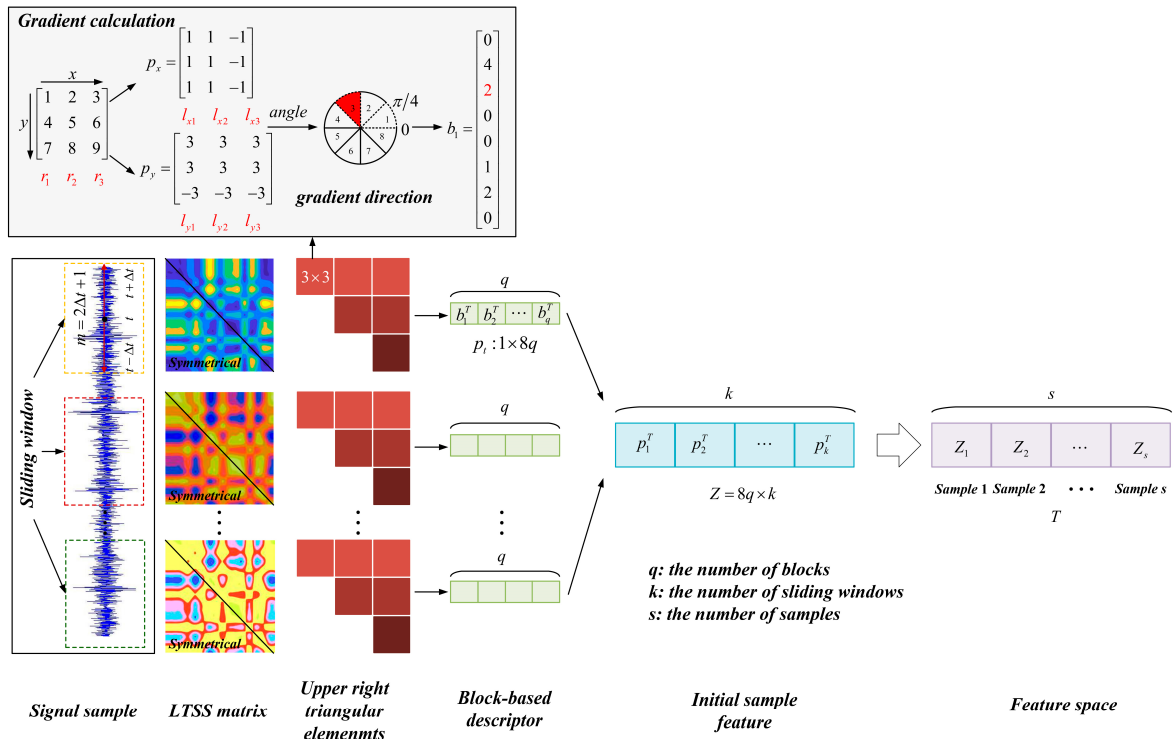


Figure 3. The flowchart of LTSS model.

Step 1: Construct the LTSS matrix from the raw signal

The raw vibration signals measured under each health condition are divided into non-overlapping and equal-length signal samples. The sliding window with a length of $m = 2\Delta t + 1$ is utilized to collect the data points around time point t from time step $t - \Delta t$ to $t + \Delta t$. The dataset $y(t)$ can be denoted as:

$$y(t) = \{x(t - \Delta t), \dots, x(t), \dots, x(t + \Delta t)\} \quad (1)$$

Then, the Euclidean distance of each two data points in $y(t)$ is calculated to construct the LTSS matrix $D(t)$:

$$D(t) = [d_{ij}] = \begin{bmatrix} d_{11} & \dots & d_{1m} \\ \vdots & \ddots & \vdots \\ d_{m1} & \dots & d_{mm} \end{bmatrix} \quad (2)$$

$$d_{ij} = \|x(i) - x(j)\| \quad (3)$$

where d_{ij} denotes the Euclidean distance between the i th element and the j th element in $y(t)$, and m is the length of $y(t)$, i.e., $2\Delta t + 1$.

Step 2: Extract the gradient feature of the LTSS matrix

Since the LTSS matrix is symmetrical, only the upper right triangular elements are considered in order to save computing resources. A block-based descriptor is employed to capture the structural information hidden in the LTSS matrix. At first, the whole matrix is divided into several blocks with a size of $n \times n$; Then, the gradient of each block is calculated to obtain column vector b_q ; Finally, all these vectors are concatenated as a multi-dimensional vector p_t , which is named as the upper right triangular block-based descriptor:

$$p_t = [b_1^T, b_2^T, \dots, b_q^T] \quad (4)$$

where T stands for transpose; q is the number of blocks.

As shown in Figure 3, the detailed procedure to calculate the gradient vector b_q is as follows. Taking a block with a size of $n \times n$ as an example, the gradient in x direction p_x is defined as:

$$p_x = [l_{x1}, l_{x2}, \dots, l_{xn}] \quad (5)$$

$$\begin{cases} l_{x1} = r_2 - r_1 \\ l_{xi} = \frac{r_{i+1} - r_{i-1}}{2} \\ l_{xn} = r_{n-1} - r_n \end{cases} \quad (6)$$

where l_{xn} is the column vector of p_x ; r_n is the column vector of the block. The similar calculation can be used to get the gradient in y direction p_y .

Then, an 8-bin histogram-based gradient direction is defined as:

$$\begin{cases} angel = \arctan\left(\frac{p_y}{p_x}\right) & (if \ p_x, p_y > 0) \\ angel = \pi + \arctan\left(\frac{p_y}{p_x}\right) & (if \ p_x < 0) \\ angel = 2\pi + \arctan\left(\frac{p_y}{p_x}\right) & (if \ p_x > 0, p_y < 0) \end{cases} \quad (7)$$

The gradient vector b_q is formed by counting the number of elements within the range of each gradient direction.

Step 3: Transform the signal sample to a sequence of LTSS feature vectors

A sequence of feature vectors p_t can be obtained by moving the sliding window with a fixed step size to traverse the entire sample signal, and repeating the above steps. Then, the signal sample can be transformed to a sequence of LTSS feature vectors as:

$$Z = [p_1^T, p_2^T, \dots, p_j^T, \dots, p_k^T] \quad (8)$$

where p_j is the feature vector extracted from the j th sliding window, and k is the number of sliding windows.

2.3.2. BoW Model

The LTSS feature extraction leads to much data redundancy, and thus brings a large computation burden. The BoW model is a simplifying assumption to construct a global representation from local features, which can be used to improve computing efficiency, making it common in many fields such as natural language processing and image/video-based action recognition [31,32]. As shown in Figure 4, for the learning phase, the BoW model performs an adaptive k-means clustering algorithm to sort all original LTSS feature vectors generating a codebook. For a new feature sample, a histogram-based encoding (HBE) strategy is used to encode it based on the codebook. The statistical feature histogram is subsequently computed as the inputs of the CapsNet layers.

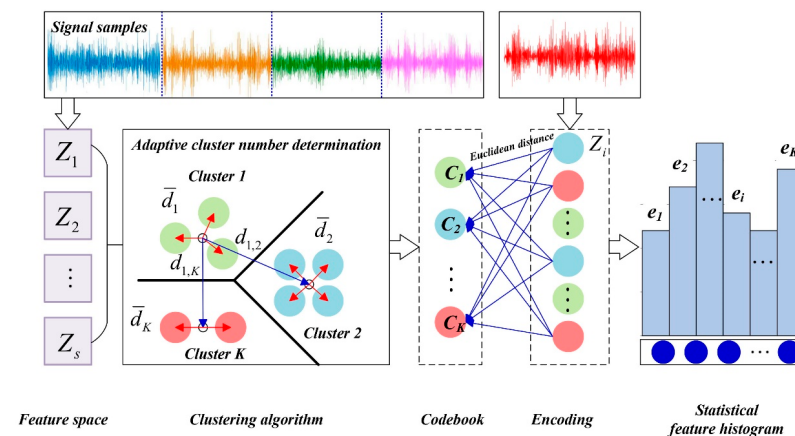


Figure 4. The flowchart of BoW model.

Step 1: Form the codebook using clustering algorithm.

Based on Section 2.3.1, the signal samples are represented as a sequence of LTSS feature vectors Z_i , and all these samples are then gathered together to form a feature space:

$$T = \bigcup_{i=1}^s Z_i \quad (9)$$

where s is the number of samples.

In this paper, the typical k -means clustering algorithm is employed to automatically learn the most representative words, i.e., codewords, which are determined by the cluster centers. Further, all these codewords form a codebook with a size of K , i.e., $\{C_i\}$, $i = 1, 2, \dots, K$, where K is the cluster number, which has great influence in clustering results.

Several approaches have been proposed to select the appropriate cluster number [35]. Among them, the Davis–Bouldin (DB) index is a promising method because of its simplicity, which is defined as the average similarity measure of each cluster with its most similar cluster, and is expressed as:

$$DB = \frac{1}{K} \sum_{i,j=1}^K \max_{i \neq j} D_{i,j} \quad (10)$$

where the similarity $D_{i,j}$ is the ratio of within-cluster distances to between-cluster distances (see Figure 4), and the expression is:

$$D_{i,j} = \frac{\bar{d}_i + \bar{d}_j}{d_{i,j}} \quad (11)$$

where $\bar{d}_i$ denotes the average distance of all points in the i -th cluster to the cluster center, $\bar{d}_j$ denotes the average distance of all points in the j -th cluster to the cluster center, $d_{i,j}$ denotes the Euclidean distance between the i -th and j -th cluster centers.

According to Equations (10) and (11), the smaller the DB index, the better the clustering results. So, the best cluster number corresponds to the minimum DB index.

Step 2: Encode the feature sample using histogram-based encoding (HBE) strategy

Histogram is an accurate representation of the distribution of numerical data, which has been widely employed in image processing, quality evaluation and time-series processing [29]. Assume that a new signal sample Y has been expressed as the LTSS feature vector form, for each point of the LTSS feature vector, the Euclidean distance between it and all the codewords in the codebook are calculated, and the codeword with minimum Euclidean distance is assigned to replace this point. Thus, the LTSS feature vector is described by a series of nearest codewords, and the frequency of each codeword is gathered to construct the statistical feature histogram, i.e., $H = \{h_1, h_2, \dots, h_K\}$.

2.4. Capsule Network for Decision-Making

The framework of CapsNet is shown in Figure 5. CapsNet usually contains a primary capsule layer and digital capsule layer. Different from the traditional neural network, the main improvements of CapsNet are: (1) traditional scalar neurons are replaced by capsule vectors to further mine the spatial information of features; (2) the dynamic routing algorithm is adopted to transmit information between the primary capsule layer and digital capsule layer, which effectively reduces the loss of feature information.

The specific parameters of CapsNet include the number and dimension of the capsules. For the primary capsule layer, the number of capsules is determined by the best cluster number K of the BoW model, and the dimension of the capsules is the same as the number of sliding windows used in the LTSS model. For the digital capsule layer, the number of digital capsules is the number of classifications, and the module length of the digital capsule vector represents the classification probability. Since the digital capsules are independent of

each other, it can predict multi-labels for test samples when making classification decisions; therefore, CapsNet is adopted as the fault classifier.

The entire process of the dynamic routing algorithm can be divided into four steps as follows. The margin loss function and average threshold adopted for decision-making are described in Sections 2.5 and 2.6, respectively.

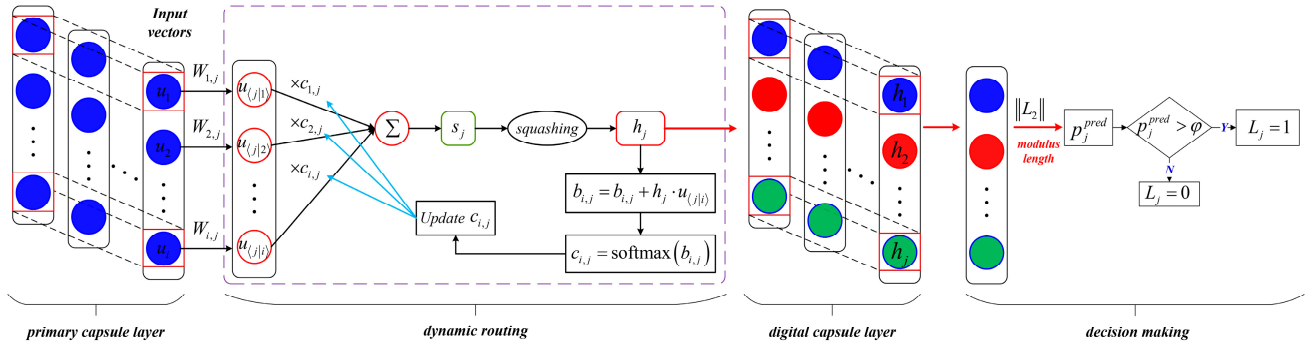


Figure 5. The framework of CapsNet.

Step 1: The input vectors u_i of the primary capsule layer are the extracted feature vectors by the previous LTSS-BoW model. Each primary capsule is multiplied by an independent weight matrix to predict the high-level capsule, which can be expressed as:

$$u_{\langle j|i \rangle} = W_{i,j} \cdot u_i \quad (12)$$

where the subscripts i and j denote the i th primary capsule and j th digital capsule, respectively. $W_{i,j}$ is the weight matrix, and $u_{\langle j|i \rangle}$ denotes the prediction vector.

Step 2: The output vector s_j is obtained by the weighted sum of all the intermediate prediction vectors $u_{\langle j|i \rangle}$, which can be expressed as:

$$s_j = \sum_i c_{i,j} \cdot u_{\langle j|i \rangle} \quad (13)$$

where $c_{i,j}$ is the coupling coefficient determined by the softmax function, which can be regarded as the connection probability that $u_{\langle j|i \rangle}$ should be coupled to s_j . The process can be expressed as:

$$c_{i,j} = \text{softmax}(b_{i,j}) = \frac{\exp(b_{i,j})}{\sum_{j=1}^k \exp(b_{i,j})} \geq 0, \sum_j c_{i,j} = 1 \quad (14)$$

where k is the number of digital capsules. $b_{i,j}$ is the prior probability of $c_{i,j}$. In the forward propagation, $b_{i,j}$ is initialized to zero and updated by dynamic routing as Algorithm 1.

Step 3: The final output vector h_j of the digital capsule layer can be obtained by the nonlinear mapping of s_j using the squashing function. The squashing function can compress the vector modulus length within the range of $[0, 1)$ without changing its orientation, which can be expressed as:

$$h_j = \text{squashing}(s_j) = \frac{\|s_j\|^2}{1 + \|s_j\|^2} \cdot \frac{s_j}{\|s_j\|} \quad (15)$$

Step 4: The dynamic routing process is executed as shown in Algorithm 1 to update $b_{i,j}$:

$$b_{i,j} = b_{i,j} + h_j \cdot u_{\langle j|i \rangle} \quad (16)$$

where the dot product $h_j \cdot u_{\langle j|i \rangle}$ is used to evaluate the similarity between the intermediate prediction vector $u_{\langle j|i \rangle}$ and the output vector h_j . The higher the similarity, the larger the values of $b_{i,j}$ and $c_{i,j}$. The optimal solution of the coupling coefficient $c_{i,j}$ can be obtained by continuous updating.

Ultimately, the final output vector h_j is returned, and the modulus length of the vector represents the classification probability p_j^{pred} .

Algorithm 1 Dynamic routing algorithm

```

1: Enter:  $u_{\langle j|i \rangle}$ 
2: Initialization parameters:  $b_{i,j}^0$ 
3: Set the number of iterations  $T$ 
4: For  $r = 1$  to  $T$  do
5:  $c_{i,j}^r = \text{softmax}(b_{i,j}^{r-1})$ 
6:  $s_j^r = \sum_i u_{\langle j|i \rangle} c_{i,j}$ 
7:  $h_j^r = \text{squashing}(s_j^r)$ 
8:  $b_{i,j}^r = b_{i,j}^{r-1} + h_j^r \cdot u_{\langle j|i \rangle}$ 
Return  $h_j$ 

Among them, softmax:  $c_{i,j} = \frac{\exp(b_{i,j})}{\sum_{j=1}^k \exp(b_{i,j})}$ 

squashing( $s_j^r$ ) =  $\frac{\|s_j^r\|^2}{1 + \|s_j^r\|^2} \cdot \frac{s_j^r}{\|s_j^r\|}$ 

```

2.5. Margin Loss Function

The margin loss function is adopted as the objective function to calculate the loss value. Compared with the cross-entropy loss function, the boundary loss function can directly measure the similarity between different classes of samples based on the Euclidean distance, which can expand the inter-class differences and reduce the intra-class differences. The expression is:

$$J = \sum_{j=1}^k L_j = \sum_{j=1}^k \left\{ T_j \max(0, m^+ - p_j^{pred})^2 + \lambda (1 - T_j) \max(0, p_j^{pred} - m^-)^2 \right\} \quad (17)$$

In the formula, k is the number of fault classes. T_j is the classification indicator function and $T_j = 1$ represents that the input sample belongs to class j , otherwise $T_j = 0$. p_j^{pred} is the predicted probability that the input sample belongs to class j . m^+ denotes the expected upper bound of the predicted probability when the sample belongs to class j . m^- denotes the expected lower bound of the predicted probability when the sample does not belong to class j . λ is the weight penalty factor.

2.6. Average Threshold

An adaptive average threshold φ is set to limit the number of output labels. If p_j^{pred} is greater than the threshold φ , the j th class label output is 1, which means the j th class exists. Otherwise, the j th class label output is 0, which means the j th class does not exist. The process can be expressed as:

$$\varphi = \text{average}(p^{pred}) = \frac{1}{k} \sum_{j=1}^k p_j^{pred} \quad (18)$$

$$L_j = 1, \quad \text{if } p_j^{pred} > \varphi \quad (19)$$

$$L_j \in L = [L_1, L_2, \dots, L_k] \quad (20)$$

where L_j is the output label of the j th class. L denotes the set of all the predicted class labels.

2.7. Diagnosis Process

Taking advantages of LTSS-BoW-based vibration feature extraction, coupled with CapsNet-based decision-making, a novel framework is proposed to diagnose the compound fault of a planetary gearbox. To summarize, the detailed steps are given below and shown in Figure 6.

- (1) Collect the vibration signals of the planetary gearbox in different health states, divide the raw signal into equal-length signal samples and normalize the data samples.
- (2) Divide the dataset into a training dataset and a test dataset. Note that the training dataset only contains the normal and single fault samples. The test dataset is composed of compound fault samples.
- (3) Design the LTSS-BoW feature extractor and convert all the samples into feature matrices.
- (4) Train the CapsNet model based on the training dataset. The trained model is used to identify the fault components of the test samples and output the predicted probability of each fault class.
- (5) Compare the predicted probability of each fault class with the average threshold for class label output.

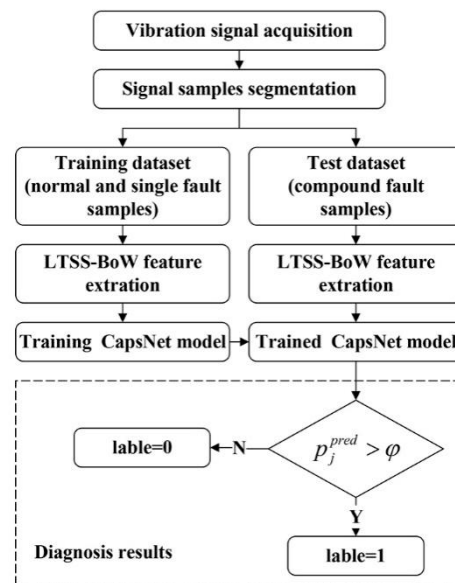


Figure 6. The diagnosis flowchart of proposed method.

3. Experimental Verification

To evaluate the effectiveness of the proposed LTSS-BoW-CapsNet diagnosis method, a series of experiments were conducted on our planetary gearbox test rig. The experimental results are analyzed in three aspects: (1) demonstrate the multi-label output results of CapsNet; (2) compare the diagnosis results of our proposed method with other methods; (3) perform feature visualization to further evaluate the feature learning ability of the proposed method on the compound fault diagnosis tasks.

3.1. Experimental Setup and Data Description

As shown in Figure 7, the test rig for a planetary gearbox consists of the drive motor, planetary reducer, magnetic powder brake, three-axis acceleration sensor installed on the gearbox and the Dewetron acquisition system. As shown in Figure 8, four kinds of single fault patterns were separately seeded on the planetary gearbox, which are sun gear tooth

crack, planet tooth crack, planet tooth surface pitting and ring gear tooth crack, denoted as SC, PC, PP and RC, respectively. Three kinds of compound fault patterns were simulated in the experiments, which are SC–PC, SC–PP and SC–RC, respectively.

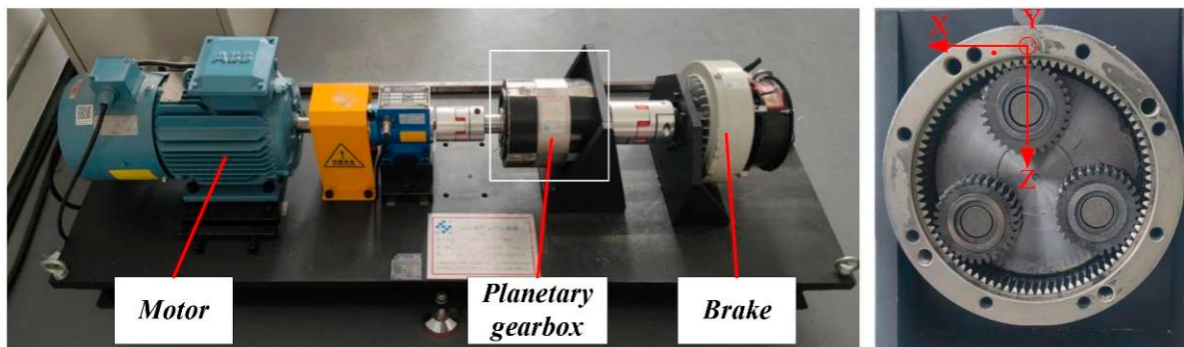


Figure 7. Planetary gearbox test rig.

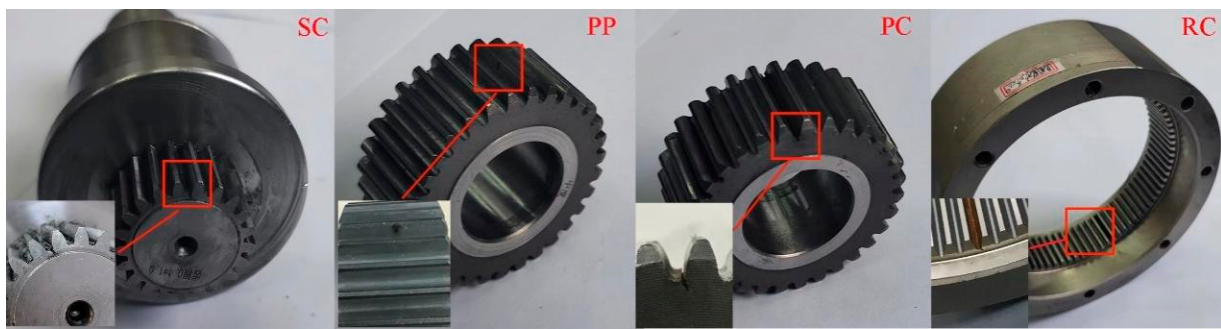


Figure 8. Gear faults.

In the experiment, the sun gear is the input component, and the carrier is the output. The rotation speed is set to 1200 rpm, and the load torque is 5 N·m. Setting the sampling frequency to 10,240 Hz, the vibration signal for each normal or faulty pattern is collected with a sampling time of 30 s. The raw signal is divided into non-overlapping signal samples and each sample has 2048 data points.

In order to reduce the impact of raw data on the diagnostic model, the normalization regularization method is adopted to normalize the raw data to between 0 and 1, and the corresponding formula is as follows:

$$M_i = \frac{N_i - \text{average}(N_i)}{\max|N_i - \text{average}(N_i)|} \quad (21)$$

Figure 9 shows the normalized time-domain signal for each fault pattern. It can be observed that the gear fault can induce impacts with a fixed period, i.e., t_s , t_p , t_r , in the time-domain signal. Compared with the single fault patterns, the coupling effect between multiple faults makes the vibration characteristics more complicated in compound fault cases. It is worth noting that new fault features occur due to the coupling effects, i.e., t_{sp} and t_{sr} ; meanwhile, the single fault-induced features are also deformed. Therefore, it is difficult to manually identify the compound fault components from the raw signal.

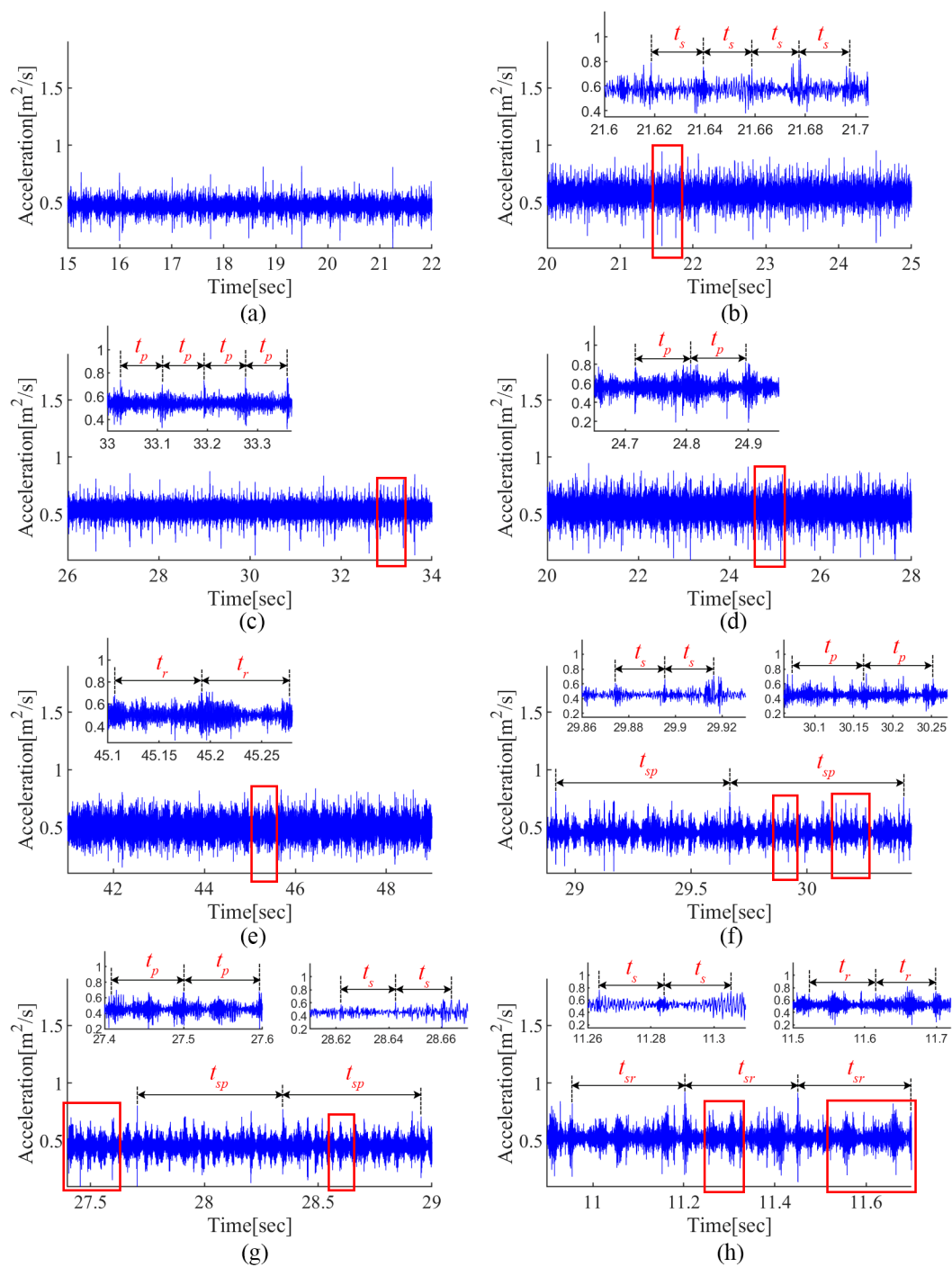


Figure 9. Normalized time-domain signal for each fault pattern (a) N, (b) SC, (c) PC, (d) PP, (e) RC, (f) SC-PC, (g) SC-PP and (h) SC-RC.

For the proposed diagnostic approach based on LTSS-BoW and CapsNet, three compound fault diagnosis tasks shown in Table 1 are set up. The normal and single fault samples are used for training based on the 5-fold cross-check method. After the model training process, the compound fault samples are used for testing. The trained model needs to predict multiple fault labels for the compound fault test samples based on the knowledge learned from the single fault samples.

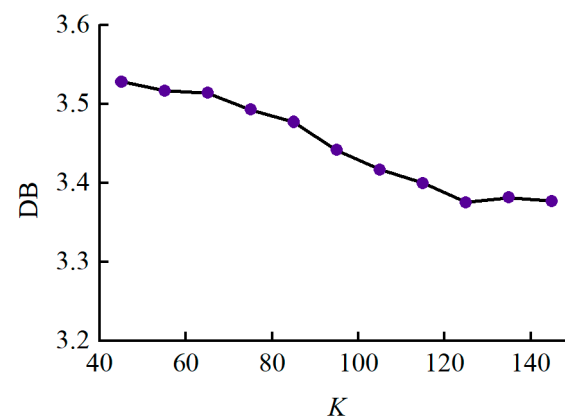
Table 1. Compound fault diagnosis tasks.

Task	Test Dataset	Training Dataset	Training Samples	Test Samples
1	SC-PC	N, SC, PC	100	10
2	SC-PP	N, SC, PP	100	10
3	SC-RC	N, SC, RC	100	10

3.2. Parameter Setting

3.2.1. Parameters of LTSS-BoW Model

The length of sliding window $m = 2\Delta t + 1$ and cluster center number K have a great influence in the calculation efficiency and accuracy. Considering the calculation complexity of the LTSS matrix, the parameter Δt is set to take value from the range $[4, 16]$ with a step size of 3. K is adaptively determined by DB index as described in Section 2.3.2. The smaller the DB index, the better the clustering results, so the best cluster number is set to 125 as shown in Figure 10. For each sample, the output matrix size is 125×5 after extracting basic features through the LTSS-BoW model.

**Figure 10.** Trend of DB index.

3.2.2. Parameters of CapsNet

The extracted features are fed into the primary capsule layer as inputs. The number of digital capsules is determined by the number of categories to be classified. During the training process, the Adam optimizer with the initial learning rate of 0.001 is adopted to update the parameters. The iteration of dynamic routing r is set to 3. The batch size is set to 10. The margin loss function is adopted to calculate the loss value. The structural parameters of the network in this paper are greatly reduced, which is more conducive to improve the training speed. The specific parameters used for the LTSS-BoW model and CapsNet are summarized in Table 2.

Table 2. Model parameters.

Layer	Parameters	Value
LTSS-BoW	Δt	4, 7, 10, 13, 16
	The cluster center number K	125
CapsNet	The number of primary capsules	125
	The dimension of primary capsules	5
	The number of digital capsules	3
	The dimension of digital capsules	10
	The iteration of dynamic routing r	3

Our approach is based on the Pytorch framework and trained on an NVIDIA RTX3070 GPU.

3.3. Diagnosis Results

3.3.1. The Predicted Probability for Multi-Label Output

For compound fault diagnosis task 1, the predicted probability values for each pattern are listed in Table 3 and shown in Figure 11. The LTSS-BoW-CapsNet model is trained based on the signal samples of normal, SC and PC patterns. Then, the trained model is tested based on the signal samples of an SC–PC compound fault pattern. In our experiments, each task was performed independently ten times to eliminate the influence of randomness. For all tests, the predicted probability values for the existence of SC and PC fault patterns are significantly higher than the average threshold, while the predicted probability values for normal patterns are far below the threshold. Therefore, the class labels of SC and PC fault patterns are equal to 1. Thus, the model accurately identifies the fault components of the SC–PC compound fault pattern and can output two single labels simultaneously. A similar procedure can be used to analyze the multi-label output results in task 2 and 3.

Table 3. The predicted probability for each pattern in task 1.

Number of Tests	Predicted Probability			Average Threshold
	SC	PC	N	
1	0.72	0.75	0.07	0.5133
2	0.77	0.61	0	0.46
3	0.74	0.7	0.01	0.4833
4	0.76	0.7	0.03	0.4967
5	0.78	0.65	0.01	0.48
6	0.73	0.74	0.07	0.5133
7	0.78	0.68	0.04	0.5
8	0.77	0.7	0.06	0.51
9	0.79	0.75	0.21	0.5833
10	0.76	0.67	0.01	0.48

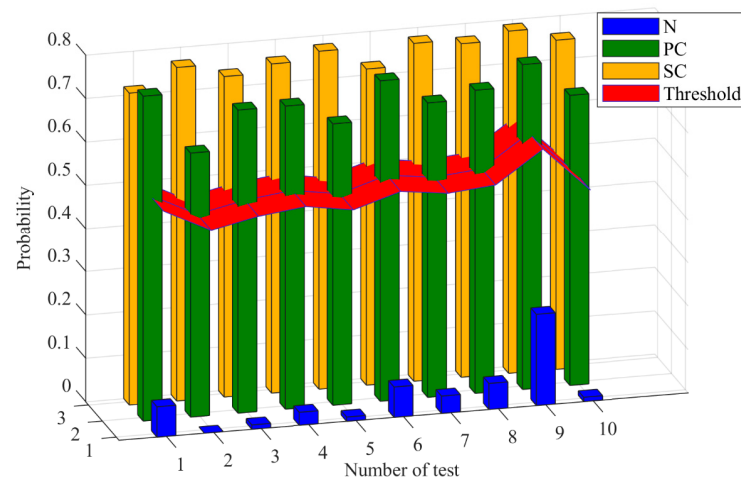


Figure 11. The predicted probability values for each pattern in task 1.

3.3.2. Comparative Analysis

To validate the effectiveness of the proposed LTSS-BoW-CapsNet method, four models were selected to compare the diagnosis performance. The comparison models include an SVM-based model, a kNN-based model, a CNN-based model and a CNN-CapsNet model, which are briefly described below:

- SVM-based and kNN-based models. To compare the effect of a classifier, two widely used classifiers SVM and kNN are used for making classification decisions. These two methods extract features based on the same LTSS-BoW model.
- CNN-based model. CNN is a typical neural network with convolution and pooling operations. The classical LetNet5 model is used here for comparison.

- (iii). CNN-CapsNet model [26]. This method uses a convolution network as a feature extractor, and a capsule network as the classifier. The parameter settings are described in Ref. [26].

In our experiments, each diagnosis task was performed independently ten times to obtain the average accuracy. Table 4 lists the accuracies of the four models on the compound fault diagnosis tasks. The results show that the SVM-based, kNN-based and CNN-based models failed in all three tasks due to the limitation of the classification principle. The CNN-CapsNet model only identifies the fault components of SC–RC and failed in the tasks SC–PC and SC–PP. The proposed LTSS-BoW-CapsNet method performed well in all tasks with an accuracy of more than 97%. This demonstrates that the proposed method can identify via decoupling the fault components and have better stability for different types of compound fault diagnosis.

Table 4. Diagnosis accuracies of all the models.

Method	Task 1	Task 2	Task 3
LTSS-BoW-SVM	0%	0%	0%
LTSS-BoW-kNN	0%	0%	0%
CNN	0%	0%	0%
CNN-CapsNet	0%	0%	100%
Our proposed method	100%	97%	100%

To further visually analyze the diagnosis results, the confusion matrices and the label outputs of four diagnosis methods (LTSS-BoW-SVM, CNN, CNN-CapsNet and the proposed LTSS-BoW-CapsNet) for three diagnosis tasks are displayed in Figures 12 and 13, respectively.

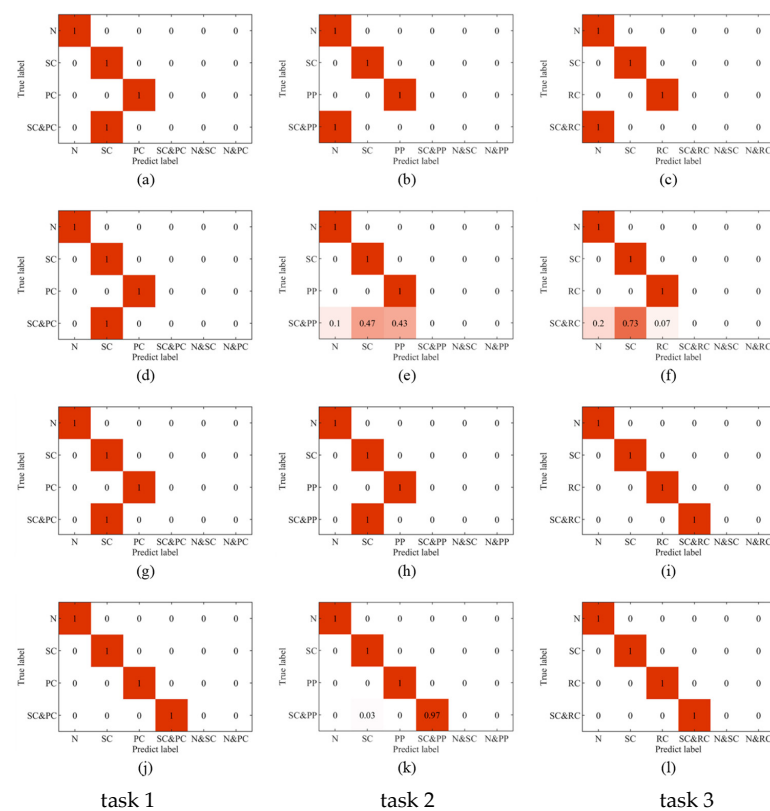


Figure 12. Confusion matrices of four diagnosis methods for three compound fault diagnosis tasks (a–c) LTSS-BoW-SVM model; (d–f) CNN model; (g–i) CNN-CapsNet model; (j–l) LTSS-BoW-CapsNet model.

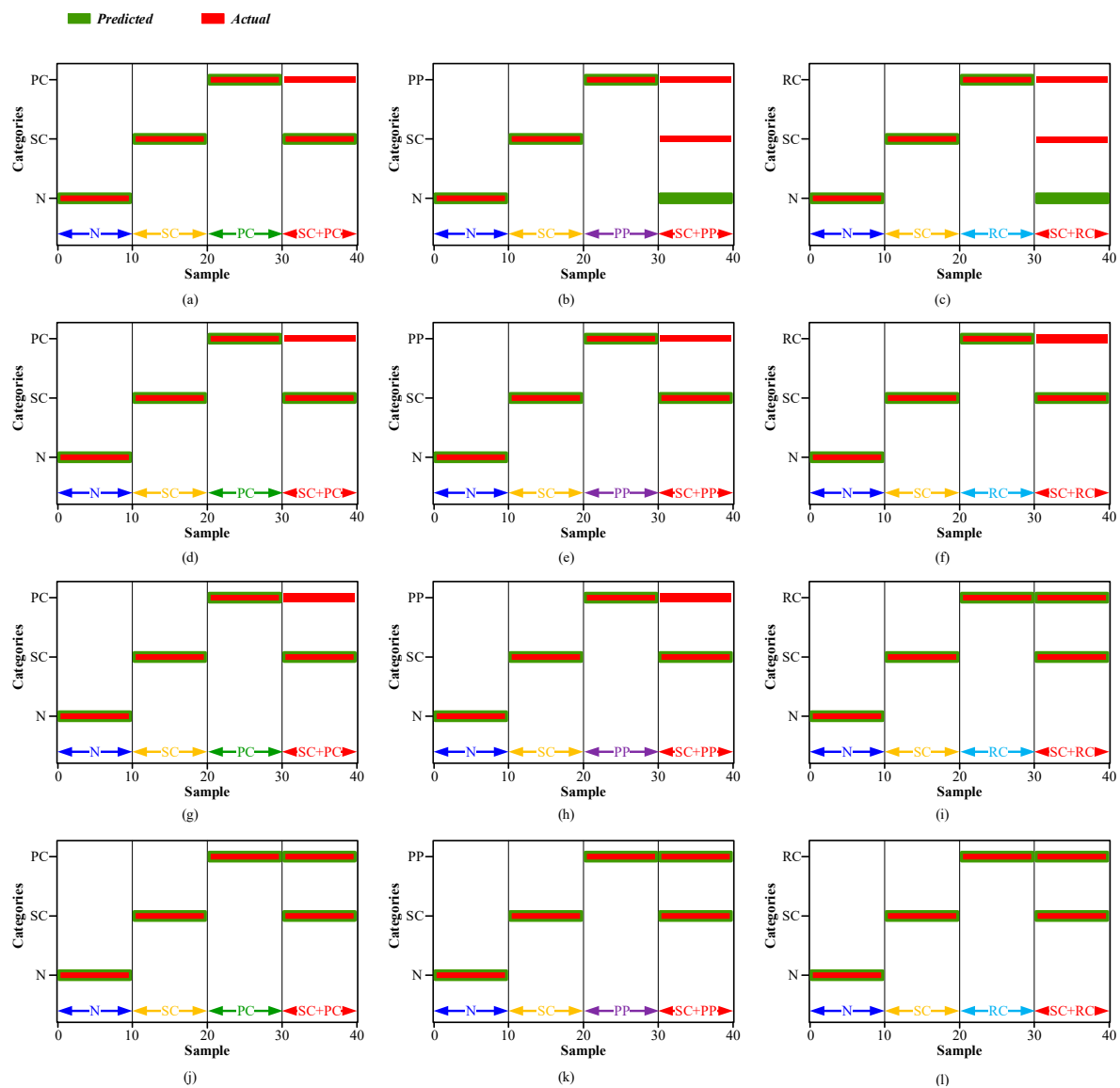


Figure 13. Label outputs of four diagnosis methods for three compound fault diagnosis tasks (a–c) LTSS-BoW-SVM model; (d–f) CNN model; (g–i) CNN-CapsNet model; (j–l) LTSS-BoW-CapsNet model.

It can be clearly seen that the LTSS-BoW-SVM and CNN models only output single class labels. The reason is that the traditional classifiers identify the most obvious features and output the most likely single fault label for compound fault diagnosis task. Therefore, the traditional classifiers cannot output multiple independent labels at the same time; it is unable to identify via decoupling the fault components contained in the compound fault.

CNN-CapsNet identifies the SC and RC fault components contained in the SC–RC compound fault, while wrongly identifying the SC–PC and SC–PP compound faults as an SC single fault. The reason could be the fault features of SC are more obvious than PC and PP faults in the compound fault signals. Therefore, the CNN has the limitation in compound fault feature extraction, especially in the case that one fault component has greater influence than the other one.

The proposed LTSS-BoW-CapsNet model successfully identifies the fault components contained in the compound faults in three tasks, which indicates that the LTSS-BoW model has better feature extraction ability. Moreover, the CapsNet model can output multi-labels

due to its unique classification principle. Above all, the proposed model has significant advantages in compound fault diagnosis.

Additionally, a t-SNE visual diagram is used to downscale the deep feature embedding and obtain the feature distribution of the CNN-CapsNet and LTSS-BoW-CapsNet models. As shown in Figure 14, comparing the feature results extracted by the high-level capsule layer in task 1, it can be seen that the class spacing between the SC–PC pattern and SC pattern is much closer than the class spacing with the PC pattern for the CNN-CapsNet method, which makes it easy to mistakenly identify the SC–PC class as SC. However, the class spacing distribution between the SC–PC pattern and two single fault patterns is uniform and clear for the LTSS-BoW-CapsNet method, so the fault components contained in the compound fault can be effectively identified. It indicates that the use of LTSS-BoW can enhance the ability of the network model to extract coupling features.

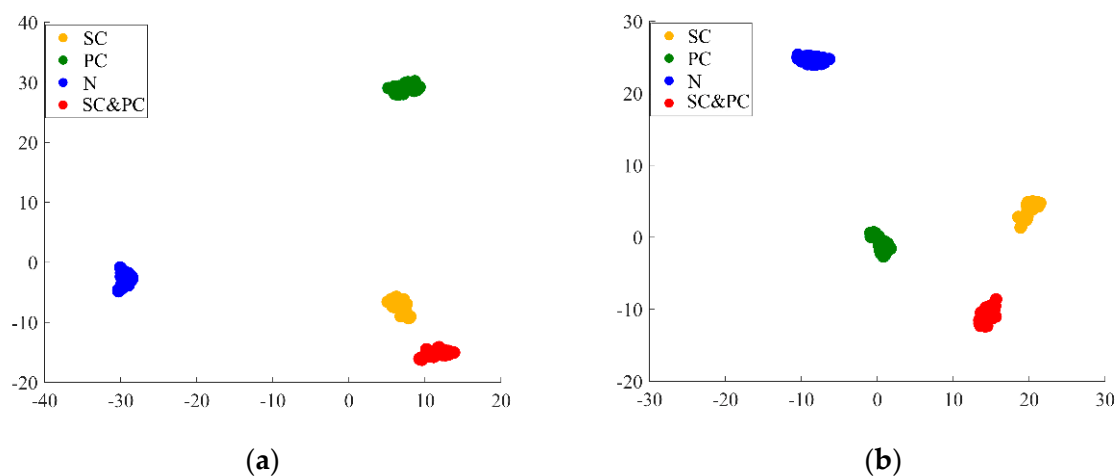


Figure 14. t-SNE visual diagrams in task 1 (a) CNN-CapsNet, (b) LTSS-BoW-CapsNet.

4. Conclusions

In this paper, a novel LTSS-BoW-CapsNet framework is proposed to diagnose the compound fault of a planetary gearbox. An improved LTSS-BoW feature extractor is constructed to extract fault feature vectors, which has the advantages of high feature extraction efficiency and strong robustness. Then, a multi-label classifier based on CapsNet is designed. The dynamic routing algorithm and average threshold are adopted to predict multi-labels for compound fault components recognition.

The effectiveness of the proposed LTSS-BoW-CapsNet method is evaluated by processing three compound fault diagnosis tasks. The experimental results demonstrate that our proposed approach can effectively identify via decoupling the multi-fault components contained in the compound fault signals of planetary gearbox. The testing accuracy is more than 97%, which is better than the other four traditional classification models. The trained model can only use the fault knowledge learned from the labeled single fault training samples to identify the fault components of compound fault test samples. Therefore, it can solve the problem that the compound fault samples are insufficient in practice.

This research only realized the diagnosis of compound faults containing two types of faults. However, the compound fault of a planetary gearbox could be more complex in practice. Therefore, in future work, the proposed method would be improved by using multi-channel signal fusion and feature fusion, so that it can identify more fault components via decoupling and achieve better identification performance.

Author Contributions: Methodology, L.H.; Software, Y.R.; Validation, X.L.; Investigation, R.L.; Data curation, J.Z.; Writing—original draft, G.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by National Natural Science Foundation of China (No. 51805354), China Postdoctoral Science Foundation (2020M673170), Natural Science Foundation of Shanxi Province (202203021221087) and Advanced Manufacturing and Intelligent Equipment Industry Research Institute of Haian-Taiyuan University of Technology.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

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

References

1. Kumar, A.; Gandhi, C.P.; Zhou, Y.; Kumar, R.; Xiang, J. Latest developments in gear defect diagnosis and prognosis: A review. *Measurement* **2020**, *158*, 107735. [\[CrossRef\]](#)
2. Chen, Y.; Zuo, M. A sparse multivariate time series model-based fault detection method for a gearbox under variable speed condition. *Mech. Syst. Signal Process.* **2022**, *167*, 108539. [\[CrossRef\]](#)
3. Xu, Y.; Feng, G.; Tang, X.; Yang, S.; Gu, F.; Ball, A.D. A modulation signal bispectrum enhanced squared envelope for the detection and diagnosis of compound epicyclic gear faults. *Struct. Health Monit.* **2023**, *22*, 562–580. [\[CrossRef\]](#)
4. Feng, Y.; Zhang, X.; Jiang, H.; Li, J. Compound fault diagnosis of a wind turbine gearbox based on MOMEDA and parallel parameter optimized resonant sparse decomposition. *Sensors* **2022**, *22*, 8017. [\[CrossRef\]](#)
5. Teng, W.; Ding, X.; Cheng, H.; Han, C.; Liu, Y.; Mu, H. Compound faults diagnosis and analysis for a wind turbine gearbox via a novel vibration model and empirical wavelet transform. *Renew. Energy* **2019**, *136*, 393–402. [\[CrossRef\]](#)
6. Zhao, Y.; Fan, Y.; Li, H.; Gao, X. Rolling bearing composite fault diagnosis method based on EEMD fusion feature. *J. Mech. Sci. Technol.* **2022**, *36*, 4563–4570. [\[CrossRef\]](#)
7. Pan, H.; Yang, Y.; Li, X.; Zheng, J.; Cheng, J. Symplectic geometry mode decomposition and its application to rotating machinery compound fault diagnosis. *Mech. Syst. Signal Process.* **2019**, *114*, 189–211. [\[CrossRef\]](#)
8. Huang, D.; Ke, L.; Mi, B.; Zhao, L.; Sun, G. A new incipient fault diagnosis method combining improved RLS and LMD algorithm for rolling bearings with strong background noise. *IEEE Access* **2018**, *6*, 26001–26010.
9. Wang, L.; Liu, Z. An improved local characteristic-scale decomposition to restrict end effects, mode mixing and its application to extract incipient bearing fault signal. *Mech. Syst. Signal Process.* **2021**, *156*, 107657. [\[CrossRef\]](#)
10. Zhang, J.; Zhang, J.; Zhong, M.; Zhong, J.; Zheng, J.; Yao, L. Detection for incipient damages of wind turbine rolling bearing based on VMD-AMCKD method. *IEEE Access* **2019**, *7*, 67944–67959. [\[CrossRef\]](#)
11. Hu, H.; Feng, F.; Jiang, F.; Zhou, X.; Zhu, J.; Xue, J.; Jiang, P.; Li, Y.; Qian, Y.; Sun, G.; et al. Gear fault detection in a planetary gearbox using deep belief network. *Math. Probl. Eng.* **2022**, *2022*, 9908074.
12. 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. [\[CrossRef\]](#)
13. Zhao, B.; Yuan, Q. Improved generative adversarial network for vibration-based fault diagnosis with imbalanced data. *Measurement* **2021**, *169*, 108522. [\[CrossRef\]](#)
14. Zhang, J.; Xu, B.; Wang, Z.; Zhang, J. An FSK-MBCNN based method for compound fault diagnosis in wind turbine gearboxes. *Measurement* **2021**, *172*, 108933. [\[CrossRef\]](#)
15. Gao, S.; Wang, X.; Miao, X.; Su, C.; Li, Y. ASM1D-GAN: An intelligent fault diagnosis method based on assembled 1D convolutional neural network and generative adversarial networks. *J. Signal Process. Syst.* **2019**, *91*, 1237–1247. [\[CrossRef\]](#)
16. Cheng, Y.; Lin, M.; Wu, J.; Zhu, H.; Shao, X. Intelligent fault diagnosis of rotating machinery based on continuous wavelet transform-local binary convolutional neural network. *Knowl.-Based Syst.* **2021**, *216*, 106796. [\[CrossRef\]](#)
17. Chen, Y.; Rao, M.; Zuo, M. Physics-informed LSTM hyperparameters selection for gearbox fault detection. *Mech. Syst. Signal Process.* **2022**, *171*, 108907. [\[CrossRef\]](#)
18. Xiao, Y.; Shao, H.; Feng, M.; Han, T.; Wan, J.; Liu, B. Towards trustworthy rotating machinery fault diagnosis via attention uncertainty in Transformer. *J. Manuf. Syst.* **2023**, *70*, 186–201. [\[CrossRef\]](#)
19. Xiao, Y.; Shao, H.; Wang, J.; Yan, S.; Liu, B. Bayesian Variational Transformer: A generalizable model for rotating machinery fault diagnosis. *Mech. Syst. Signal Process.* **2024**, *207*, 110936. [\[CrossRef\]](#)
20. Sabour, S.; Frosst, N.; Hinton, G.E. Dynamic routing between capsules. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017.
21. Fan, C.; Xie, H.; Tao, J.; Li, Y.; Pei, G.; Li, T.; Lv, Z. ICaps-ResLSTM: Improved capsule network and residual LSTM for EEG emotion recognition. *Biomed. Signal Process. Control* **2024**, *87*, 105422. [\[CrossRef\]](#)
22. Yang, M.; Zhao, W.; Chen, L.; Qu, Q.; Zhao, Z.; Shen, Y. Investigating the transferring capability of capsule networks for text classification. *Neural Netw.* **2019**, *118*, 247–261. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Sezer, A.; Sezer, H.B. Capsule network-based classification of rotator cuff pathologies from MRI. *Comput. Electr. Eng.* **2019**, *80*, 106480. [\[CrossRef\]](#)

24. Liu, J.; Zhang, C.; Jiang, X. Imbalanced fault diagnosis of rolling bearing using improved MsR-GAN and feature enhancement-driven CapsNet. *Mech. Syst. Signal Process.* **2022**, *168*, 108664. [[CrossRef](#)]
25. Li, D.; Zhang, M.; Kang, T.; Li, B.; Xiang, H.; Wang, K.; Pei, Z.; Tang, X.; Wang, P. Fault diagnosis of rotating machinery based on dual convolutional-capsule network (DC-CN). *Measurement* **2022**, *187*, 110258. [[CrossRef](#)]
26. Liang, P.; Deng, C.; Yuan, X.; Zhang, L. A deep capsule neural network with data augmentation generative adversarial networks for single and simultaneous fault diagnosis of wind turbine gearbox. *ISA Trans.* **2023**, *135*, 462–475. [[CrossRef](#)]
27. Xu, Q.; Liu, C.; Yang, E.; Wang, M. An improved convolutional capsule network for compound fault diagnosis of RV reducers. *Sensors* **2022**, *22*, 6442. [[CrossRef](#)]
28. Huang, R.; Liao, Y.; Zhang, S.; Li, W. Deep decoupling convolutional neural network for intelligent compound fault diagnosis. *IEEE Access* **2019**, *7*, 1848–1858. [[CrossRef](#)]
29. Huang, R.; Li, J.; Li, W.; Cui, L. Deep ensemble capsule network for intelligent compound fault diagnosis using multisensory data. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 2304–2314. [[CrossRef](#)]
30. Junejo, I.N.; Dexter, E.; Laptev, I.; Pérez, P. Cross-View action recognition from temporal self-similarities. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 172–185. [[CrossRef](#)] [[PubMed](#)]
31. Yang, S.; Lu, G.; Wang, A.; Liu, J.; Yan, P. Change detection in rotational speed of industrial machinery using Bag-of-Words based feature extraction from vibration signals. *Measurement* **2019**, *146*, 467–478. [[CrossRef](#)]
32. Peng, X.; Wang, L.; Wang, X.; Qiao, Y. Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice. *Comput. Vis. Image Underst.* **2016**, *150*, 109–125. [[CrossRef](#)]
33. Nie, Y.; Li, F.; Wang, L.; Li, J.; Wang, M.; Sun, M.; Li, G.; Li, Y. Phenomenological vibration models of planetary gearboxes for gear local fault diagnosis. *Mech. Mach. Theory* **2022**, *170*, 104698. [[CrossRef](#)]
34. Li, G.; Liang, X.; Li, F. Model-based analysis and fault diagnosis of a compound planetary gear set with damaged sun gear. *J. Mech. Sci. Technol.* **2018**, *32*, 3081–3096. [[CrossRef](#)]
35. Kodinariya, T.M.; Makwana, P.R. Review on determining number of cluster in k-means clustering. *Int. J. Adv. Res. Comput. Sci. Manag. Stud.* **2013**, *1*, 90–95.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.