Signals hierarchical feature enhancement method for CNN-based fault diagnosis

Huang Zhang, Shuyou Zhang, Zili Wang, Lemiao Qiu and Yiming Zhang

Abstract
The high noise and low energy characteristics of the raw signals collected by sensors make the signal features weak and difficult to train. The purpose of this paper is to enhance the fault features of abnormal signals using the hierarchical feature enhancement method (HFE) which contains three layers. In the first layer, the signals are decomposed into multiple modals estimated by a variational optimization problem. The modals we choose are used to reconstruct the signals to form a complex matrix used to extract features in the second layer. In the third layer, the feature signals are converted into two-dimensional space and then are input into the convolutional neural network (CNN) for fault diagnosis after HFE since CNN helps to mine deeper features and compute in parallel on a large scale. The experimental results effectively verify the performance of the HFE for enhancing the weak fault features and preventing noise interference. The signals analyzed by HFE used as input greatly improve the diagnosis ability of CNN. In addition, the ablation and comparison experiments are conducted which still show superiority.

Keywords
Hierarchical feature enhancement method (HFE), convolutional neural network (CNN), fault diagnosis

Date received: 20 June 2022; accepted: 15 August 2022

Handling Editor: Chenhui Liang

Introduction
Fault information will be reflected in the equipment through signals, and these fault messages often form abnormal signals that are difficult to monitor through external equipment. Indirect sensing methods diagnose faults by analyzing condition monitoring data collected via sensors including temperature, forces, vibration, and sound. Hence, indirect sensing methods are more practical while eliminating the need for manual inspection intervention making the detection process more efficient and less costly. Fault diagnosis methods can be broadly divided into two categories: model-based methods and data-driven methods. Model-based methods measure process variables and estimate residuals based on the physical properties of the dynamic system. Increasingly, a large amount of sensor data drives the development of data-driven methods. These approaches focus on the condition monitoring data which means no prior knowledge of the data distribution is required. It makes the data-driven approach more agreeable.

Signals are constantly collected during operation to accumulate large amounts of historical data, making the data-driven approach increasingly accurate. Among

The State Key Laboratory of Fluid Power and Mechatronic System, Zhejiang University, Hangzhou, China

Corresponding author:
Zili Wang, The State Key Laboratory of Fluid Power and Mechatronic System, Zhejiang University, Room-416, Buiding-1st, Yuquan Campus, Hangzhou, Zhejiang 310027, China.
Email: ziliwang@zju.edu.cn

Creative Commons CC BY: This article is distributed under the terms of the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/) which permits any use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access pages (https://us.sagepub.com/en-us/nam/open-access-at-sage).
those methods, the deep learning methods are widely used in recent years. Since a deep belief network is applied to fault detection, more and more scholars use the deep learning method for fault diagnosis. Since the data is generally one-dimensional, LSTM and GRU methods are commonly used for diagnosis. However, LSTM and GRU too rely on the current state. Meanwhile, the inability to process multiple temporal data in parallel makes LSTM and GRU increasingly disadvantageous in fault diagnosis. Compared with the situation, we found that it is the CNN model used in learning and analyzing that can be processed in parallel on a large scale, which has played a more and more essential role in image processing and signal processing as one of the most core algorithms in deep learning. CNN could reflect the characteristics of the signals on a deeper level and has great advantages in parameter sharing and sparse connectivity which accelerate the speed of computation and suppress the model overfitting. Hence, CNN is used more and more in signal processing which is helpful in diagnosis. However, although it can get good results in processing, one-dimensional CNN can be affected by noise easily. Then 2-D CNN is adopted since 2-D CNN has excellent feature extraction capability. Lu et al. combined the GA algorithm with 2-D CNN to make full use of 2-D CNN’s advantages in image processing. Therefore, converting one-dimensional fault signals into two-dimension is helpful for fault identification and classification.

The fault signals collected have the characteristics of non-linearity and non-stationarity. At the same time, due to the large noise generated during the operation of the equipment, the signals are easily drowned in the noise, making the fault features hard to extract and difficult to process. Although CNN has the ability to learn features by itself to some extent, features cannot be extracted well due to the fixed length of convolutional kernel length. To solve this problem, feature engineering is used as an auxiliary tool to better help CNN extract multi-domain features which means the features must be enhanced to draw the fault features better for fault diagnosis.

In the current research on signal feature enhancement, wavelet packet decomposition, empirical modal decomposition, and feature enhancement techniques based on entropy are mostly used. Among these three methods, wavelet packet decomposition extracts the feature parameters of different frequency bands and weighted summation of the feature parameters to approximate and segment the original signal in detail. The empirical modal decomposition decomposes the signal to estimate the similarity or cumulative contribution of each component to select representative features from the components. The representative method based on the principle of minimum entropy is the minimum entropy deconvolution which improves the fault feature extraction by enhancing the impact pulse of the signal. However, the effectiveness of wavelet packet decomposition is based on the selection of wavelet basis. Although it can provide a window that can adaptively adjust with the change of signal frequency, it improves time accuracy by sacrificing frequency accuracy. The empirical modal decomposition and the minimum entropy deconvolution suffer from edge effects leading to distortion.

Currently, most of the fault diagnosis is performed by enhancing the signals through the above-mentioned enhancement methods and then directly using the classification model SVM or combining it with the one-dimensional neural network models RNN, LSTM to determine the faults. However, the deeper dimension signals are difficult to extract in one dimension and contain less information which makes models hard to diagnose.

In this paper, we propose a hierarchical feature enhancement method to enhance the features of fault signals. The HFE contains three layers including a signal denoising layer, a feature extraction layer, and a feature transforming layer. The signal denoising layer decomposes the signals into multiple modals to prevent noise interference. In the feature extraction layer, a Gaussian window function is added to the series to form a complex matrix for feature extraction. Then the feature transforming layer converts the feature sequence into polar coordinates from one-dimensional signals to two-dimensional space. CNN is used after the HFE to diagnose the faults.

The main innovations and contributions of our work are as follows:

1. The three layers in the HFE perform their respective duties. Fault features submerged in noise can be better extracted after the HFE.
2. The features extracted by the HFE can effectively resist the interference of noise, and can still better enhance the fault features in high noise scenarios.
3. Using the advantages of two-dimensional CNN in extracting high-dimensional features, the fault features are further enhanced, which greatly improves the accuracy of model diagnosis. At the same time, the experiments before and after the HFE show that the HFE plays a pivotal role in improving the accuracy of the model.

The remaining part of this paper is as follows: methods including HFE and CNN are introduced in Section 2. The experimental verification and ablation experiments are conducted and the results are also compared with other CNN-based feature enhancement methods.
in Section 3. Additionally, the conclusions and future work are summarized in Section 4.

**Method**

HFE method contains three layers which are the signal denoising layer, feature extraction layer, and feature transforming layer. HFE is used before CNN to separate the signals from noise and enhance the fault features. After HFE, signals have been converted into images that include enhanced features. And then those images are input into the CNN to train and test in order to realize the aim of fault diagnosis. The framework of the proposed signals HFE method for CNN-based fault diagnosis is shown in Figure 1.

**Hierarchical feature enhancement method**

**Signal denoising layer.** The fault features are buried in the noise and the feature information cannot be extracted well due to the fixed length of convolutional kernel length. HFE method is adopted to better enhance the features of signals with faults.

The signal denoising layer is mainly used to separate the signals from noise. The sampled signals contain noise which can be expressed as follows

\[
f_k = f + \eta = \sum_k u_k
\]

Where \(\eta\) is noise and \(f\) is the signal after denoising. The modal decomposition is used to decompose the signals into several modals.

To find \(f\), the regularization method is used.

\[
\min \|\sum_k u_k - f\|_2^2 + \alpha \sum_k \|f_k\|_2^2
\]

Where \(k\) is the mode number of decomposition. The first part of (2) is the cost function. Due to the non-uniqueness of the solution during decomposition, this problem is ill-posed. Therefore, the meaning of the second part is to add norms to eliminate ill-posed. L2 regular terms are added, \(\alpha\) is the penalty factor of L2 regular term, which is used to control the degree of
penalty. The value is adjusted through the test set of subsequent experiments.\textsuperscript{18}

Then the update formula for alternating direction update technique is adopted after that.

\[ J[\hat{u}_1, \hat{u}_2, \ldots, \hat{u}_k] = \int \left( \hat{f}(\omega) - \hat{u}_k(\omega) \right)^2 + \alpha \omega^2(\hat{u}_k(\omega))^2 \, d\omega \]

(3)

Then the extreme value of can be found through (3).

To search for the optimal central frequency \( \omega_k \), the alternating direction update technique is adopted after the update formula for \( \hat{u}_k \). Introduce the multiplier \( \lambda \) thereby converting the constrained variational expression into an unconstrained variational expression.

\[ J = \int \left( \hat{f}(\omega) - \sum_k \hat{u}_k(\omega) \right)^2 + \frac{\lambda(\omega)}{1 + \alpha(\omega - \omega_k)^2} + \int \lambda(\omega) \hat{f}(\omega) - \sum_k \hat{u}_k(\omega)) \, d\omega \]

(4)

With the same progress, the final update equations are obtained as follows through calculating.

\[ \hat{f}(\omega) - \sum_k \hat{u}_k(\omega) + \frac{\lambda(\omega)}{1 + \alpha(\omega - \omega_k)^2} \]

(5)

\[ \omega_k = \frac{\int_{0}^{\infty} \omega(\hat{u}_k(\omega))^2 \, d\omega}{\int_{0}^{\infty} (\hat{u}_k(\omega))^2 \, d\omega} \]

(6)

\[ \hat{\lambda}^{n+1}(\omega) = \hat{\lambda}^n(\omega) - \tau(\hat{f}(\omega) - \sum_k \hat{u}_k(\omega)) \]

(7)

The decomposition progress is based on variational mode decomposition.\textsuperscript{18} \( \tau \) is the noise tolerance, the default of it is 0.001. The iteration progress is stopped through precision \( \varepsilon \) which is small enough but bigger than 0.

\[ \sum_k \left\| \hat{u}_k^{n+1} - \hat{u}_k^n \right\|^2_2 < \varepsilon \]

(8)

**Feature extraction layer.** After performing the decomposition, the modal components to be retained are selected and the components to be rejected are rounded off. The signals after decomposition is \( f_k - u \), \( u \) is the modal to be removed. And then Fourier transform is performed on the signals after the signal denoising layer, and a Gaussian window function is added to the signals.

Since the height and width of the Gaussian window function vary with frequency, the drawbacks caused by the fixed window width are improved, allowing the transform to adjust the resolution.

\[ F(f) = \int_{-\infty}^{\infty} (f_k - u)e^{-2\pi f^2} \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{f^2}{\sigma^2}} \, dt \]

(9)

Scale and translate the Gaussian window functions. Then the transform can be expressed as (10).

\[ S(f, \tau) = \int_{-\infty}^{\infty} (f_k - u) \left| f \right| e^{-\frac{f^2}{\sigma^2}} e^{-2\pi f^2} \, dt \]

(10)

Where \( \sigma = \frac{1}{\sqrt{\tau}} \).

Through (10), a two-dimensional matrix can be obtained, where the columns correspond to the sampling time, and the rows correspond to the frequency values. The matrix elements are complex values from which the amplitude and phase information can be extracted. The features representing the signals are extracted from the two-dimensional matrix, and the features extracted are then standardized as \( F = [F_1, F_2, \ldots, F_m] \).\textsuperscript{19} Where \( F_i = [F_{i1}, F_{i2}, \ldots, F_{im}] \).

Where \( n \) is the number of features extracted and \( m \) is the number of input signals.

\[ F = \begin{bmatrix} F_{11} & F_{12} & \cdots & F_{1n} \\ F_{21} & F_{22} & \cdots & F_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ F_{m1} & F_{m2} & \cdots & F_{mn} \end{bmatrix} \]

(11)

Calculate its correlation coefficient matrix.

\[ R = \frac{1}{m-1} F^T F = [r_{ij}] = \frac{1}{m-1} \sum_{k=1}^{m} F_{ik} F_{jk} (i, j = 1, 2, \ldots, n) \]

(12)

N non-negative eigenvalues are obtained through (12) which are \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N \geq 0 \). A matrix is formed from the eigenvectors corresponding to the eigenvalues.

\[ A = \begin{bmatrix} \xi_{11} & \xi_{12} & \cdots & \xi_{1N} \\ \xi_{21} & \xi_{22} & \cdots & \xi_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \xi_{n1} & \xi_{n2} & \cdots & \xi_{nN} \end{bmatrix} \]

(13)

Each column in \( A \) is the eigenvector corresponding to \( \lambda_i \). Select the first \( k \) eigenvalues according contribution rate. And the calculation of contribution rate is:

\[ \frac{\lambda_i}{\sum_{j=1}^{k} \lambda_j} \]

(14)

The normalized \( F \) is multiplied by the eigenvector matrix corresponding to the selected eigenvalues to reconstruct the new features.
Feature transformation layer. Rearrange the regenerated features into a one-dimensional sequence to generate new eigenvectors.

Assume the series is $\alpha = (\alpha_1, \alpha_2, ..., \alpha_n)$ through feature extraction layer. And in the next feature transforming layer, signals are normalized first so that all values fall between $[-1, 1]$.

$$\tilde{\alpha}_i = \frac{(\alpha_i - \max(\alpha)) + (\alpha_i - \min(\alpha))}{\max(\alpha) - \min(\alpha)}, \quad -1 \leq \tilde{\alpha}_i \leq 1$$  \hspace{1cm} (15)

Transform the sequence into polar coordinates after normalization.\(^{20}\)

$$\begin{cases} 
\phi = \arccos(\tilde{\alpha}_i) \\
\rho = \frac{\tilde{\alpha}_i}{N}, t_i \in \mathbb{N}
\end{cases}$$  \hspace{1cm} (16)

Where $t_i$ is time stamp, and $N$ is a constant to adjust the polar span.

Thus, the correlation matrix $G$ is formed.

$$G = \begin{pmatrix} 
\cos(\varphi_1 + \varphi_1) & \cos(\varphi_1 + \varphi_2) & \cdots & \cos(\varphi_1 + \varphi_n) \\
\cos(\varphi_2 + \varphi_1) & \cos(\varphi_2 + \varphi_2) & \cdots & \cos(\varphi_2 + \varphi_n) \\
\vdots & \vdots & \ddots & \vdots \\
\cos(\varphi_n + \varphi_1) & \cos(\varphi_n + \varphi_2) & \cdots & \cos(\varphi_n + \varphi_n)
\end{pmatrix}$$  \hspace{1cm} (17)

Correlations in different time intervals can be identified by triangulation and normalization. Each element of $G$ is the cosine value of the sum that adds the corresponding position to the angles of other positions. Through the above steps, one-dimensional signals can be converted into two-dimensional space.

To sum up, one-dimensional signals are converted into two-dimensional space through three layers: signal denoising layer, feature extraction layer, and feature transforming layer. The signal denoising layer reduces the noise from the raw vibration signals. And features that can represent the signals are extracted through the feature extraction layer. The feature transforming layer converts the signals into higher space, so the higher features can be shown and extracted better. The transformation layer can reflect the fault characteristics more intuitively, and the fault characteristics after the HFE analysis have been enhanced, which makes the fault easier to be diagnosed.

Fault classification method

CNN was chosen as a fault diagnosis method to judge and classify various fault errors. Compared with other methods, CNN could evidently reflect the characteristics on a deeper level and has the advantages of parameters sharing and local connectivity achieving higher accuracy. CNN received the features enhanced by the HFE as input, and the features enhanced by the HFE have been converted to two-dimensional space. CNN can process the signals in parallel which is suitable for image processing and for use in our work. The convolution operation process can be expressed as follows:

$$X^k_j = f\left(\sum_{i \in M_j} X^{k-1}_i W^k_j + b^k_j\right), j = 1, ..., N$$  \hspace{1cm} (18)

Where $M_j$ is the $j$ convolution area in the $k$ layer, $k$ is the layer of the network, $X^k_j$ is the $j$ output in the $k$ layer, $X^{k-1}_i$ is the $i$ input in the $k$ layer, $W^k_j$ is the weight matrix of the convolution kernel, and $b^k_j$ is the bias, $f$ is the activation function which is ReLU commonly. ReLU is the activation function that is used to implement a nonlinear projection of the input. ReLU defines the nonlinear output result. It helps to speed up the convergence and prevent the gradient from vanishing.

$$X^k_j_{ReLU} = f(X^k_j) = \max(0, X^k_j)$$  \hspace{1cm} (19)

The Pooling layer is a form of down-sampling. It can effectively reduce the size of the parameter matrix and accelerate the computational speed. After the initial feature extraction of the convolution layer, the pooling layer is further used to extract features. As one of them, the max-pooling layer divides the input images into several rectangular regions and outputs the maximum value of each sub-region.

$$X^k_j_{Pooling} = \max(\text{window}(l_1, l_2) \cap X^k_j_{ReLU})$$  \hspace{1cm} (20)

Where $\text{window}(l_1, l_2)$ is the pooling window which can slide along the side with certain steps. $l_1, l_2$ is the dimension of the window. $\cap$ means the sliding window coincides with the field of view.

After convolution and pooling operations, features are then input to the fully-connected layer which are flattened into a vector. The main goal of the fully-connected layer is to further extract features and then combine the output with a softmax classifier. The probability of output $y_i$ corresponding to the category label set to $i$ is determined.

$$y_i = \frac{\exp(w_ix + b_i)}{\sum_{i=1}^{N} \exp(w_ix + b_i)}$$  \hspace{1cm} (21)

Where $i$ represents a category in $N$.

Experiment

To verify the effects of the HFE method for CNN-based fault diagnosis, the bearing datasets are selected for learning and analysis as the bearing is the most
basic and critical part of electromechanical equipment and its signals are easily affected by noise.

**MFPT data set**

The experimental data were obtained from the bearing failure fault sets provided by Machinery Failure Prevention Technology (MFPT), whose acceleration bearing at 270 lb load and a sampling frequency of 97,656 Hz. The two different bearing faults are shown in Figure 2 and the parameters of the bearings are listed in Table 1.

The experimental data contain vibration signals under normal conditions, as well as inner race faults, outer race faults under different loads, and three real-world faults including an intermediate shaft bearing from a wind turbine, an oil pump shaft bearing from a wind turbine, and a real-world planet bearing fault. The bearings in the faulty conditions in the experiment were subjected to seven different levels of load. The details are shown in Table 2.

To analyze the MFPT datasets entirely, we treated the datasets in two categories: manual simulation faults including faults at different locations, under seven different loads, and three real-world bearing faults.

### HFE enhancing

The signals are easily disturbed by external interference in the test rig, making the collected signals mixed with noise and causing the fault diagnosis difficult.

The signal denoising layer decomposes the signals into K modals. Although the parameter $\alpha$ is adjusted by the test set, it will not affect the experimental results too much. Set penalty factor which is also bandwidth constraint $\alpha = 2000$ as a common value. From equations (5)–(7), the expression of each modal and the iteration of the central frequency can be obtained. The K value is chosen through the instantaneous frequency. The signals are decomposed into K modals and the jth sampling point’s instantaneous frequency of the ith modal can be expressed as $f_{ij}$. Perform decomposition on the signals based each of K to obtain modal components $\hat{u}_k(\omega)$. And through inverse Fourier transform $\hat{u}_k(\omega)$ can be transformed into time domain to get $\hat{u}_k(t)$.

$$\hat{u}_k(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{u}_k(\omega)e^{ij\omega t} d\omega$$

Then analytic signals can be obtained through the Hilbert transform.

$$\hat{z}_k(t) = \hat{u}_k(t) + j\hat{y}_k(t)$$

Where

$$\hat{y}_k(t) = \hat{u}_k(t) * h(t)$$

$$h(t) = 1/\pi t$$

* is the convolution operation. The instantaneous frequency is obtained by deriving the instantaneous phase of $\hat{z}_k(t)$ from time.

### Table 1. Bearing parameters of MFPT.

| Roller diameter (mm) | Pitch diameter (mm) | Number of elements | Contact angle (°) |
|---------------------|---------------------|--------------------|------------------|
| 5.969               | 31.623              | 8                  | 0                |

### Table 2. Data set composition of MFPT.

| Fault source     | Fault type                  | Sampling frequency (Hz) | Sampling duration (s) | Load (lbs)       |
|------------------|-----------------------------|-------------------------|-----------------------|------------------|
| Manual intervention | Normal                      | 97,656                  | 8                     | 270              |
|                  | Inner race fault            | 48,828                  | 3                     | 50,100,150,200,250,300 |
|                  | Outer race fault            | 48,828                  | 3                     | 2,550,100,150,200,250,300 |
| Real world fault | Intermediate shaft bearing  | 48,828                  | 10                    | 300              |
|                  | Oil pump shaft bearing      | 24,414                  | 20                    | 300              |
|                  | Real world planet bearing fault | 6104                   | 40                    | /                |
Then the mean of instantaneous of $f_{ij}$ can be expressed as

$$f_i = \frac{1}{N} \sum_j f_{ij}$$  \hspace{1cm} (26)

$N$ is the number of instantaneous frequencies in one modal.

According to the instantaneous frequency mean value method, the variation curve of the instantaneous frequency of signal data with $K$ value can be obtained. We choose the oil pump bearing to show the trend of $K$ in Figure 3.

If the $K$ value is too large, the number of decomposition is too large, and the instantaneous jump phenomenon is severe. And the sudden jump will raise the mean value which will cause bending.

From Figure 3, the high-frequency component of the transient frequency jump raises the transient frequency average of the modal component, so the transient frequency average curve is bent suddenly at $K = 13$, so for oil pump bearing, $K$ is 13. Similarly, $K$ for the intermediate bearing is 8, and $K$ for planet bearing is 6.

The $K$ value is determined as 13, then the decomposition is performed by $K$ to obtain the different modals. Different modals are shown in Figure 4.

From Figure 4, the first modal contains the most noise. Therefore the first modal is removed to eliminate high-frequency noise. Then the rest of the modal components are summed up to reconstruct the signals. The feature extraction layer adds a Gaussian window function instead of a wavelet basis function to the Fourier transform to obtain frequency information and amplitude information of signals. Then a complex matrix is obtained. To extract the features, the modular matrix is obtained by modular operation, and the time-frequency (TF) contour of two-dimensional images is obtained in Figure 5. From the TF curve in Figure 5, it is clear that the energy accumulation range of different fault radius frequencies is different.

To better extract the features of bearing data with different fault categories, a total of 11 features are chosen according to the modulus matrix: skewness, kurtosis, peak to peak value, maximum frequency, the standard deviation of frequency, root mean square, maximum value, minimum value, average value, the average of absolute value and variance.

Although 11 features are extracted, the weights of the 11 features are different for fault signals. Dimensionality reduction of features is conducive to selecting the most important and independent features, and deleting redundant features reduces the amount of computation. Therefore, the signals are downscaled to generate new features to reduce computation.

Through formula (14), the contribution rate can be obtained, and the result is shown in Figure 6. From Figure 6, it can be seen that the first three characteristics account for the majority, and the total contribution of the first three features is 94.9342%. So 95% contribution rate is chosen as the selection criterion which leads to three eigenvectors for generating new features.

For the bearing without fault, the new features are:

- **Principle 1:**
  
  \[0.3604F_1 + 0.3692F_2 + 0.0604F_3 + 0.1144F_4 + 0.3507F_5 + 0.3692F_6 + 0.0029F_7 + 0.3129F_8 + 0.3691F_9 + 0.3129F_{10} + 0.3546F_{11}\]

- **Principle 2:**
  
  \[0.0195F_1 + 0.1706F_2 + 0.5023F_3 + 0.5945F_4 - 0.2611F_5 + 0.1706F_6 - 0.0212F_7 - 0.3439F_8 + 0.1711F_9 - 0.3439F_{10} + 0.0347F_{11}\]

- **Principle 3:**
  
  \[-0.1128F_1 + 0.0156F_2 + 0.1932F_3 - 0.0208F_4 + 0.0301F_5 + 0.0156F_6 + 0.9602F_7 + 0.0795F_8 - 0.0095F_9 + 0.0795F_{10} - 0.1165F_{11}\]

---

**Figure 3.** Instantaneous frequency-K value diagram.

**Figure 4.** K modals of oil pump bearing.
For the intermediate bearing fault, the new features are:

\[
\text{Principle1} = 0.3584F_1 + 0.3603F_2 + 0.0511F_3 + 0.1464F_4 + 0.3538F_5 \\
+ 0.3603F_6 + 0.0294F_7 + 0.3236F_8 + 0.3603F_9 + 0.3236F_{10} + 0.3501F_{11} \\
\text{Principle2} = 0.0235F_1 + 0.0977F_2 + 0.6438F_3 + 0.6092F_4 - 0.1617F_5 \\
+ 0.0977F_6 + 0.0129F_7 - 0.2818F_8 + 0.0976F_9 - 0.2818F_{10} + 0.0090F_{11} \\
\text{Principle3} = -0.0180F_1 - 0.0158F_2 - 0.0132F_3 + 0.0065F_4 - 0.0047F_5 \\
- 0.0158F_6 + 0.9990F_7 + 0.0050F_8 - 0.0271F_9 + 0.0050F_{10} - 0.0104F_{11}
\]

For the oil pump bearing fault, the new features are:

\[
\text{Principle1} = 0.3625F_1 + 0.3539F_2 - 0.0538F_3 + 0.0288F_4 + 0.3657F_5 \\
+ 0.3539F_6 + 0.0487F_7 + 0.3374F_8 + 0.3537F_9 + 0.3374F_{10} + 0.3541F_{11} \\
\text{Principle2} = 0.0060F_1 + 0.2479F_2 + 0.5220F_3 + 0.5980F_4 - 0.2034F_5 \\
+ 0.2479F_6 - 0.0273F_7 - 0.2676F_8 + 0.2484F_9 - 0.2676F_{10} + 0.0046F_{11} \\
\text{Principle3} = -0.0542F_1 - 0.0048F_2 + 0.0622F_3 + 0.0111F_4 - 0.0061F_5 \\
- 0.0048F_6 + 0.9940F_7 + 0.0124F_8 - 0.0145F_9 + 0.0124F_{10} - 0.0659F_{11}
\]

For the planet bearing fault, the new features are:

\[
\text{Principle1} = 0.3587F_1 + 0.3599F_2 + 0.0282F_3 + 0.1287F_4 + 0.3657F_5 \\
+ 0.3539F_6 + 0.0487F_7 + 0.3374F_8 + 0.3537F_9 + 0.3374F_{10} + 0.3541F_{11} \\
\text{Principle2} = 0.0282F_1 + 0.1034F_2 + 0.6445F_3 + 0.6209F_4 - 0.1493F_5 \\
+ 0.1034F_6 - 0.0429F_7 - 0.2662F_8 + 0.1036F_9 - 0.2662F_{10} + 0.0174F_{11} \\
\text{Principle3} = 0.0035F_1 + 0.0182F_2 + 0.0319F_3 + 0.0462F_4 + 0.0116F_5 \\
+ 0.0182F_6 + 0.9976F_7 + 0.0181F_8 + 0.0105F_9 + 0.0181F_{10} - 0.0024F_{11}
\]

Figure 5. Time-frequency contour.
Then new features are spliced together to form new features. Converting the one-dimensional signals into two-dimensional space helps to further analyze the characteristics of faults and thus classify different faults. After regenerating new features from the matrix, the new one-dimensional vector is reconstructed. After that, the signals are dynamically transformed into polar coordinates to form two-dimensional feature pictures. Since all the values are in the same scale of $[2, 1, 1]$, every pixel in the picture can be expressed as different colors corresponding to values according to equation (17).

**Fault classification**

The labels of MFPT dataset corresponding to seven different loads are 0, 1, 2, 3, 4, 5, and 6 respectively. Labels corresponding dissimilar real-world faults are 0, 1, 2, and 3 respectively. The bearing data under seven different loads are first used for training. The CNN structure is designed concerning the VGG structure, and a simplification is made based on it after the HFE method to diagnose faults. The structure is shown in Figure 7. Since the CNN structure is used to further extract deeper features and classify the different fault features enhanced by HFE. The results of the analysis are mainly to compare the impact of the adoption of HFE on the classification results of CNN, so the CNN structure is not the main. Simplified analysis based on VGG structure can not only achieve the purpose of comparison but also improve the calculation speed.

The picture size is set to $128 \times 128$ pixels. Therefore, training picture datasets are formed in each fault state. The fault classification problem for different loads of the MFPT dataset is a seven classification problem for the loads that have seven different states. The three real-world faults plus the data without faults can be counted as a four-category discriminant. For different defects, characteristic values are different, the defect characteristic diagrams are also not the same. The CNN structure is used to learn the features on a deeper level. The network structure is simplified to reduce the computation and speed up the training.

A total of nine convolution layers and three Max-pooling layers are connected to learn the characteristic maps. Since the pictures that contain faults are hard to classify, it needs several convolution layers to learn the features. And the max-pooling layers are after the convolution layers to further obtain the optimal features. To classify the different types of faults, the features are flattened, and fully-connected layers integrate the feature representation in the next to output the result as a value. Then dropout is chosen to avoid over-fitting. An activation function of ReLU is selected in every layer.

The CNN structure is designed concerning the VGG structure, and a simplification is made based on it after the HFE method to diagnose faults. The structure is shown in Figure 7. Since the CNN structure is used to further extract deeper features and classify the different fault features enhanced by HFE. The results of the analysis are mainly to compare the impact of the adoption of HFE on the classification results of CNN, so the CNN structure is not the main. Simplified analysis based on VGG structure can not only achieve the purpose of comparison but also improve the calculation speed.

The picture size is set to $128 \times 128$ pixels. Therefore, training picture datasets are formed in each fault state. The fault classification problem for different loads of the MFPT dataset is a seven classification problem for the loads that have seven different states. The three real-world faults plus the data without faults can be counted as a four-category discriminant. For different defects, characteristic values are different, the defect characteristic diagrams are also not the same. The CNN structure is used to learn the features on a deeper level. The network structure is simplified to reduce the computation and speed up the training.

A total of nine convolution layers and three Max-pooling layers are connected to learn the characteristic maps. Since the pictures that contain faults are hard to classify, it needs several convolution layers to learn the features. And the max-pooling layers are after the convolution layers to further obtain the optimal features. To classify the different types of faults, the features are flattened, and fully-connected layers integrate the feature representation in the next to output the result as a value. Then dropout is chosen to avoid over-fitting. An activation function of ReLU is selected in every layer.

The CNN structure is designed concerning the VGG structure, and a simplification is made based on it after the HFE method to diagnose faults. The structure is shown in Figure 7. Since the CNN structure is used to further extract deeper features and classify the different fault features enhanced by HFE. The results of the analysis are mainly to compare the impact of the adoption of HFE on the classification results of CNN, so the CNN structure is not the main. Simplified analysis based on VGG structure can not only achieve the purpose of comparison but also improve the calculation speed.

The picture size is set to $128 \times 128$ pixels. Therefore, training picture datasets are formed in each fault state. The fault classification problem for different loads of the MFPT dataset is a seven classification problem for the loads that have seven different states. The three real-world faults plus the data without faults can be counted as a four-category discriminant. For different defects, characteristic values are different, the defect characteristic diagrams are also not the same. The CNN structure is used to learn the features on a deeper level. The network structure is simplified to reduce the computation and speed up the training.

A total of nine convolution layers and three Max-pooling layers are connected to learn the characteristic maps. Since the pictures that contain faults are hard to classify, it needs several convolution layers to learn the features. And the max-pooling layers are after the convolution layers to further obtain the optimal features. To classify the different types of faults, the features are flattened, and fully-connected layers integrate the feature representation in the next to output the result as a value. Then dropout is chosen to avoid over-fitting. An activation function of ReLU is selected in every layer.
convolution layer and fully-connected layer. Choose epoch of 1500 and batch size of 100 for the model to train. The proportion of the test set is set to 33%, dropout is 0.25 and the weight decay is $10^{-5}$, then the prediction discrimination confusion matrix is obtained in Figure 8 which is trained by MFPT datasets. Inner race and outer race bearing under seven dissimilar loads results and three real-world bearing faults with normal bearing results are shown in Figure 8.

From the confusion matrix of MFPT in Figure 8, we can get when the epoch is set to 1500 and the batch size is 100, the model, used to distinguish the seven different loads of inner and outer race bearing faults has dissimilar accuracy when recognizing faults. Not all the accuracies are high. But the model used to distinguish three real-world bearing faults with one normal bearing is performing well. The reason for the difference in recognition accuracy of different models is related to the parameters of model training.

In order to evaluate the training effect of the model, the evaluation index of learning is selected as: precision, recall, f1 and loss.

When it is a four-categories classification problem, the correct classification and prediction results can be expressed in Table 3, the accurate classification samples are distributed on the diagonal.

Generally, n classes are calculated as N binary tasks, and each class is processed and filled in separately. For A, the calculating progress is shown in Table 4.

From Table 4 we can calculate the prediction of A which is:

$$P = \frac{TP}{TP + FP} = \frac{AA}{AA + (AB + AC + AD)}$$  \hspace{1cm} (27)

Then we can also calculate following the same progress to get $\bar{P}$.

$\bar{P} = \frac{PA + PB + PC + PD}{4}$  \hspace{1cm} (28)

The value of recall is:

$$R = \frac{TP}{TP + FN} = \frac{AA}{AA + (BA + CA + DA)}$$  \hspace{1cm} (29)

Then we can also calculate the same progress to get $\bar{R}$.

$$\bar{R} = \frac{RA + RB + RC + RD}{4}$$  \hspace{1cm} (30)

The value of $F_A$:

$$F_A = \frac{2P_AR_A}{P_A + \bar{R}_A}$$  \hspace{1cm} (31)

Then $F$ can be obtained.

$$F = \frac{FA + FB + FC + FD}{4}$$  \hspace{1cm} (32)

Loss:

$$CE(x) = -\sum_{i=1}^{C} y_i \log f_i(x)$$  \hspace{1cm} (33)

Where $x$ is the input sample, $C$ is the total number of categories to be classified, $y_i$ is the real label of the ith category and $f_i(x)$ is the output values for corresponding models.

For batch, the loss of all samples in the batch is averaged.

$$CE(x)_{\text{final}} = \frac{\sum_{b=1}^{N} CE(x^{(b)})}{N}$$  \hspace{1cm} (34)

The calculation of precision, recall, f1, and loss is illustrated by the four classification tasks.
After training, four metric values can be obtained about the model under seven different loads in the MFPT data sets. For the outer race bearing, the four indexes are precision: 0.9787, loss: 0.0690, f1: 0.977, and recall: 0.975. For the inner race bearing the indexes are: precision: 0.9780, loss: 0.0733, f1: 0.9754, recall: 0.9729. For real-world bearing failures the indexes are precision: 0.9921, loss: 0.0266, f1: 0.9920, recall: 0.9918.

Ablation experiments and analysis

Some parameters will affect the training accuracy of the model, such as the batch size of the training set. To further improve the fault classification accuracy of the model, the batch size is adjusted. We use four learning evaluation indexes (precision, recall, f1, and loss) that are mentioned above to compare the model training ability under different batch sizes. The obtained training results for different loads on the inner and outer race bearing of the MFPT datasets are shown in Table 5.

It can be seen from Table 5 that when the batch size is 64, the results have converged basically when the epoch equals 1000. In order to further illustrate the effect of HFE on model training when the batch size and epoch are changed. Batch sizes equal 64 and equal 128 are often used in network training. Since when batch size equals 64 the model has achieved good results, we took two more batch sizes which are 100 and 200 as parameters in experiments to illustrate the classification effect of the CNN model before and after signal enhanced by HFE. Since a good fault diagnosis accuracy has been achieved when epoch equals 1000 in Table 5, two groups of epoch values are taken up and down respectively for experiments based on this epoch value. At the same time, in order to illustrate the extreme situation, another group of epochs equal 100 is taken for experiments to see the effect of HFE on feature enhancement and the help of CNN classification in the case of training underfitting. From Tables 6 and 7, we can get the conclusion that the HFE can improve the accuracy of the model to some extent even though the parameters of the CNN can have an impact on the training results.

In Table 6, it can be seen that the accuracy increases significantly when batch size equals 100, from 36.15% to 97.79%. And even when batch size equals 200 which causes a low accuracy, the HFE method can also improve the classification accuracy from 32.39% to 99.63%. In these two tables, the training accuracy of the model is greatly reduced when the batch size is too big or the epoch is too small. However, even in this

### Table 3. Label and prediction result representation.

| Label | A   | B   | C   | D   |
|-------|-----|-----|-----|-----|
|      | AA  | AB  | AC  | AD  |
|      | BA  | BB  | BC  | BD  |
|      | CA  | CB  | CC  | CD  |
|      | DA  | DB  | DC  | DD  |

### Table 4. Label and prediction result representation of A.

| Label | Prediction = A | Prediction = non A |
|-------|----------------|-------------------|
| Label | TP(A) = AA     | FP(A) = AB + AC + AD |
|       | FN(A) = BA + CA + DA | TN(A) = BB + BC + BD |
|       | CB + CC + CD  | DB + DC + DD    |
|       | DB             |                  |
In this case, the accuracy can still be improved greatly by applying the HFE method. In hence, the HFE method can greatly enhance the characteristics of faults and improve the accuracy of fault diagnosis. To more intuitively display the feature enhancement effect of HFE on the signals in different noise environments, we intercepted a section of the signal and added different Gaussian noises, which are -5, -10, 5, and 10 dBW respectively. The results are shown in Figure 9. It can be seen from Figure 9 that the region with the largest time-domain signal oscillation on the left in Figure 9(a) forms the feature formed by line crossing in the middle feature map, and the line crossing feature is enhanced after feature enhancement by HFE. By adding different Gaussian noise to the original signal, the time-domain signal in Figure 9(b)–(e) can be formed. The features after adding noise become unclear which is hard to diagnose, but the features become obvious after processing through HFE. Therefore, HFE can enhance the fault features even in high noise scenarios.

With the development of neural network methods, more and more scholars use CNN for fault diagnosis and classification. The comparative results of CNN with different enhancement methods are shown in Table 8. It can be seen that many scholars have achieved good results in classification for MFPT datasets whose accuracy of fault classification is up to 98.80% according to STCNN proposed by Li et al. Our classification also achieves good results which are higher than 99%. However, the HFE method proposed by us only combines with a simple CNN structure for analysis. Therefore, the HFE method proposed by us can greatly enhance the fault features and improve the accuracy of the classification model.

**Conclusion and future work**

Signals are easily drowned in high noise which makes features hard to extract and train. Based on this, the

---

### Table 5. Effects of batch size parameter on experiment results.

| Index | Batch size = 64 | Batch size = 100 |
|-------|-----------------|------------------|
| **Outer Precision & Loss** | | |
| Training and test precision | Training and test loss | Training and test precision | Training and test loss |
| **F1 & Recall** | | |
| Training and test f1 | Training and test recall | Training and test f1 | Training and test recall |
| **Accuracy** | 99.63% | 97.79% |

| **Inner Precision & Loss** | | |
| Training and test accuracy | Training and test loss | Training and test accuracy | Training and test loss |
| **F1 & Recall** | | |
| Training and test f1 | Training and test recall | Training and test f1 | Training and test recall |
| **Accuracy** | 99.42% | 97.41% |
HFE method is proposed for the CNN-based fault diagnosis. HFE contains three layers including a signal denoising layer, feature extraction layer, and feature transforming layer. Through HFE, the weak fault features buried in high noise are enhanced and can prevent noise interference. Then CNN is used to mine deeper features and diagnose the faults.

To verify the effectiveness of the proposed HFE method, datasets of bearings are chosen as an example since bearing signals are easily affected by the noise. Through training, the model can efficiently distinguish faults under different conditions including bearing faults under different loads and locations, artificial faults, and real faults. The accuracies of the MFPT

| Table 6. Comparison results with and without HFE about batch size. |
|---------------------------------------------------------------|
| Epoch = 1500; Parameter: batch size                           |
|                                                              |
|                                                              |
| **Batch size = 64**                                          |
|                                                              |
| **Without HFE**                                              |
|                                                              |
| **With HFE**                                                 |
|                                                              |
| **Precision & Loss**                                         |
|                                                              |
|                                                              |
| **F1 & Recall**                                              |
|                                                              |
|                                                              |
| **Confusion matrix**                                         |
|                                                              |
|                                                              |
| **Accuracy**                                                 |
|                                                              |
| 95.23%                                                      |
| 99.63%                                                      |
| 36.15%                                                      |
| 97.79%                                                      |
|                                                              |
| **Batch size = 128**                                         |
|                                                              |
| **Without HFE**                                              |
|                                                              |
| **With HFE**                                                 |
|                                                              |
| **Precision & Loss**                                         |
|                                                              |
|                                                              |
| **F1 & Recall**                                              |
|                                                              |
|                                                              |
| **Confusion matrix**                                         |
|                                                              |
|                                                              |
| **Accuracy**                                                 |
|                                                              |
| 46.25%                                                      |
| 98.04%                                                      |
| 16.88%                                                      |
| 32.39%                                                      |
Table 7. Comparison results with and without HFE about epoch.

| Batch size = 64; Parameter: epoch | Epoch = 100 | Epoch = 500 | Epoch = 1000 | Epoch = 1500 |
|----------------------------------|-------------|-------------|---------------|--------------|
| Without HFE                     | With HFE    | Without HFE | With HFE      | Without HFE | With HFE |
| **Precision & Loss**             |             |             |               |             |         |
| Training and test precision      | Training and test precision | Training and test precision | Training and test precision | Training and test precision |
| Training and test loss           | Training and test loss | Training and test loss | Training and test loss | Training and test loss |
| **F1 & Recall**                  |             |             |               |             |         |
| Training and test F1             | Training and test F1 | Training and test F1 | Training and test F1 | Training and test F1 |
| Training and test Recall         | Training and test Recall | Training and test Recall | Training and test Recall | Training and test Recall |
| **Confusion matrix**             |             |             |               |             |         |
| Confusion matrix of Outer race bearing without enhancing | Confusion matrix of Outer race bearing with enhancing | Confusion matrix of Outer race bearing without enhancing | Confusion matrix of Outer race bearing with enhancing |
| **Accuracy**                     | 14.78%      | 18.38%↑     | 28.85%        | 59.87%↑     |

| Precision & Loss | F1 & Recall | Confusion matrix | Accuracy |
|------------------|-------------|------------------|----------|
| Training and test precision | Training and test precision | Training and test precision | Training and test precision | Training and test precision |
datasets are great which are all above 99% in different conditions. At the same time, signals in different noise scenarios were simulated and transformed into two-dimensional images to intuitively show features that fully illustrate the robustness of the model.

Although the model used in this paper achieves good results, the CNN used has been simplified without considering the specific design of the model in depth. Therefore, the structure of the model needs to be further considered in the future, maybe using some relevant optimization algorithms. With that, the proposed HFE can be combined with a more optimized network model for fault classification and diagnosis.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work has been funded by the National Natural Science Foundation of China (51905476), the Public Welfare Technology Application Projects of Zhejiang Province, China (LGG22E050008), the Jiangsu Province Science and Technology Achievement Transforming Fund Project (BA2018083).

**ORCID iDs**

Zili Wang https://orcid.org/0000-0003-2198-534X

Lemiao Qiu https://orcid.org/0000-0001-9358-0099

**References**

1. Liu H, Liu Z, Jia W, et al. Tool wear estimation using a CNN-transformer model with semi-supervised learning. *Meas Sci Technol* 2021; 32: 125010.
2. He X, Qu J, Zhong T, et al. Diagnosis intermittent fault of bearing based on EEMD-HMM. In: *International conference on mechatronics and intelligent robotics*, 2018, pp.174–182. Cham: Springer.
3. Jiang Y, Yin S and Kaynak O. Data-driven monitoring and safety control of industrial cyber-physical systems: basics and beyond. *IEEE Access* 2018; 6: 47374–47384.
4. Tamilselvan P and Wang P. Failure diagnosis using deep belief learning based health state classification. *Reliab Eng Syst Saf* 2013; 115: 124–135.
5. Cao Y, Ding Y, Jia M, et al. A novel temporal convolutional network with residual self-attention mechanism for remaining useful life prediction of rolling bearings. *Reliab Eng Syst Saf* 2021; 215: 107813.
6. Lyu C, Huo Z, Cheng X, et al. Distributed optical fiber sensing intrusion pattern recognition based on GAF and CNN. *J Lightwave Technol* 2020; 38: 4174–4182.
7. Zhang W, Peng G, Li C, et al. A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals. *Sensors* 2017; 17: 425.
8. Lu W, Mao H, Lin F, et al. Recognition of rolling bearing running state based on genetic algorithm and
convolutional neural network. *Adv Mech Eng* 2022; 14: 16871402110729.

9. Tang H, Liao Z, Chen P, et al. A novel convolutional neural network for low-speed structural fault diagnosis under different operating condition and its understanding via visualization. *IEEE Trans Instrum Meas* 2021; 70: 1–11.

10. Yu G, Wang X and Da C. Compound mechanical fault diagnosis based on CMDE. *Adv Mech Eng* 2022; 14: 1687132210805.

11. Wang C, Liu C, Liao M, et al. An enhanced diagnosis method for weak fault features of bearing acoustic emission signal based on compressed sensing. *Math Biosci Eng* 2021; 18: 1670–1688.

12. Medina R, Macancela JC, Lucero P, et al. A LSTM neural network approach using vibration signals for classifying faults in a gearbox. In: *International conference on sensing, diagnostics, prognostics, and control (SDPC)*, Beijing, China, 15–17 August 2019, pp.208–214. New York, NY: IEEEEE.

13. Sawalhi N, Randall RB and Endo H. The enhancement of fault detection and diagnosis in rolling element bearings using minimum entropy deconvolution combined with spectral kurtosis. *Mech Syst Signal Process* 2007; 21: 2616–2633.

14. Xu Y, Cai Z, Cai X, et al. An enhanced multipoint optimal minimum entropy deconvolution approach for bearing fault detection of spur gearbox. *J Mech Sci Technol* 2019; 33: 2573–2586.

15. Srimani S, Ghosh K and Rahaman H. Wavelet transform based fault diagnosis in analog circuits with SVM classifier. In: *IEEE international test conference India*, Bangalore, India, 12–14 July 2020, pp.1–10. New York, NY: IEEEEE.

16. Yang R, Huang M, Lu Q, et al. Rotating machinery fault diagnosis using long-short-term memory recurrent neural network. *IFAC-PapersOnLine* 2018; 51: 228–232.

17. Zhu D, Song X, Yang J, et al. A bearing fault diagnosis method based on L1 regularization transfer learning and LSTM deep learning. In: *IEEE international conference on information communication and software engineering (ICICSE)*, Chengdu, China, 19–21 March 2021, pp.308–312. New York, NY: IEEEEE.

18. Dragomiretskiy K and Zosso D. Variational mode decomposition. *IEEE Trans Signal Process* 2014; 62: 531–544.

19. Xia Y, Wang W, Song Z, et al. Fault diagnosis of flexible production line machining center based on PCA and ABC-LVQ. *Proc IMechE, Part B: J Engineering Manufacture* 2021; 235: 594–604.

20. Wang Z, Oates T. Imaging time-series to improve classification and imputation. *arXiv:1506.00327*, 2015.

21. Machinery fault prevention technology bearing data, https://mfpt.org/fault-data-sets/ (2012).

22. Deng L, Zhang A and Zhao R. Intelligent identification of incipient rolling bearing faults based on VMD and PCA-SVM. *Adv Mech Eng* 2022; 14: 16871402110729.

23. Gai J, Shen J, Hu Y, et al. An integrated method based on hybrid grey wolf optimizer improved variational mode decomposition and deep neural network for fault diagnosis of rolling bearing. *Measurement* 2020; 162: 107901.

24. Simonyan K and Zisserman A. Very deep convolutional networks for large-scale image recognition. *Comput Sci* 2014.

25. Li G, Deng C, Wu J, et al. Sensor data-driven bearing fault diagnosis based on deep convolutional neural networks and S-transform. *Sensors* 2019; 19: 2750.

26. Verstraete D, Ferrada A, Droguett EL, et al. Deep learning enabled fault diagnosis using time-frequency image analysis of rolling element bearings. *Shock Vib* 2017; 2017: 1–17.

27. San Martín G, López Droguett E, Meruane V, et al. Deep variational auto-encoders: A promising tool for dimensionality reduction and ball bearing elements fault diagnosis. *Struct Health Monit* 2019; 18: 1092–1128.

28. Yu H, Wang K and Li Y. Multiscale representations fusion with joint multiple reconstructions autoencoder for intelligent fault diagnosis. *IEEE Signal Process Lett* 2018; 25: 1880–1884.