Convolutional neural network with Huffman pooling for handling data with insufficient categories: A novel method for anomaly detection and fault diagnosis

Yuqing Li¹, Mingjia Lei¹✉, Yao Cheng², Rixin Wang¹ and Minqiang Xu¹
¹Deep Space Exploration Research Center, Harbin Institute of Technology, Harbin, China
²Beijing Institute of Spacecraft System Engineering, Beijing, China

Abstract

The rotating component is an important part of the modern mechanical equipment, and its health status has a great impact on whether the equipment can safely operate. In recent years, convolutional neural network has been widely used to identify the health status of the rotor system. Previous studies are mostly based on the premise that training set and testing set have the same categories. However, because the actual operating conditions of mechanical equipment are complex and changeable, the real diagnostic tasks usually have greater diversity than the pre-acquired datasets. The inconsistency of the categories of training set and testing set makes it easy for convolutional neural network to identify the unknown fault data as normal data, which is very fatal to equipment health management. To overcome the above problem, this article proposes a new method, Huffman-convolutional neural network, to improve the generalization ability of the model in detection task with various operating conditions. First, a new Huffman pooling kernel is designed according to the Huffman coding principle and the Huffman pooling layer structure is introduced in the convolutional neural network to enhance the model’s ability to extract common features of data under different conditions. Second, a new objective function is proposed based on softmax loss, intra-class loss, and inter-class loss to improve the Huffman-convolutional neural network’s ability to distinguish different classes of data and aggregate the same class of data. Third, the proposed method is tested on three different datasets to verify the generalization ability of the Huffman-convolutional neural network in diagnosis tasks.

Corresponding author:
Mingjia Lei, Deep Space Research Center, Harbin Institute of Technology, Building 2B, Harbin Institute of Technology Science Park, No. 2, Yikuang Street, Xiangfang District, Harbin, Heilongjiang, 150001, China. Email: hitleimingjia@163.com

Creative Commons Non Commercial CC BY-NC: This article is distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 License (https://creativecommons.org/licenses/by-nc/4.0/) which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is attributed as specified on the SAGE and Open Access page (https://us.sagepub.com/en-us/nam/open-access-at-sage).
with multi-operating conditions. Compared with other traditional methods, the proposed method has better performance and greater potential in multi-condition fault diagnosis and anomaly detection tasks with inconsistent class spaces.

Keywords
Rotating component, fault diagnosis, anomaly detection, convolutional neural network, Huffman pooling, mixed loss function

Introduction

The continuous development of modern industrial technology has prompted more and more large-scale, complicated, integrated, and intelligent equipment to emerge. The above characteristics of modern industrial equipment not only makes higher operational reliability, safety, and economy become urgent needs, but also create new challenges for equipment condition monitoring and maintenance management.\(^1−^3\) The fault diagnosis and anomaly detection technology developed under this background performs a vital function in ensuring the safe and sustainable operation of modern mechanical systems.

Fault diagnosis or anomaly detection is essentially a pattern recognition process that establishes a mapping of the measurement or feature space to the fault space. Generally, fault diagnosis methods can be divided into four categories: model-based method, signal-based method, knowledge-based method, and hybrid method.\(^4−^6\)

Traditional identification processes typically rely on a variety of ancillary maps of the monitoring signals and are ultimately diagnosed by a diagnostician. For example, the spectrum of the vibration signal, the envelope spectrum, and the demodulation spectrum are applied to diagnose the failure of the rotating machinery.\(^7−^11\)

Knowledge-based method is a very typical automatic fault identification method that requires a large amount of historic data to establish the system fault modes without prior known models or signal patterns. With the advent of the era of big data, knowledge-based fault diagnosis methods have been flourishing.\(^12,13\) The data-driven fault diagnosis method represented by machine learning (ML) and deep learning (DL) is especially typical. The vibration signals are projected into a nonlinear space by methods such as locally linear embedding,\(^14\) sparse representation,\(^15\) autoencoder,\(^16\) and support vector machines (SVMs)\(^17\) to obtain more significant features. Deep belief network (DBN),\(^18\) convolutional neural network (CNN),\(^19\) sparse autoencoder,\(^20\) stacked denoising autoencoder,\(^21\) sparse filtering,\(^22\) and other DL methods can improve the accuracy of recognition without manual feature extraction.

The fault diagnosis methods based on ML and DL introduced above are all based on a common assumption that the class labels of the test data and training data are consistent.\(^23,24\) However, in the actual fault diagnosis task, different working conditions and even new fault types will lead to inconsistency between the test data and the training data category.\(^25\) With the in-depth study of intelligent diagnosis technology, in the field of rotating equipment fault diagnosis, researchers’ attention on the diagnosis problem has gradually shifted from the consistent category space to the inconsistent category space. The category space inconsistency in the fault diagnosis problem mainly includes equipment inconsistency, working condition inconsistency, and fault class
inconsistency. From this, techniques such as open-set fault diagnosis (OSFD)\textsuperscript{26} and cross-domain fault diagnosis (CDFD)\textsuperscript{27} are derived.

The OSFD is mainly used to solve fault diagnosis problems in which there are more fault classes in test data than in training data. This requires the diagnostic model to not only correctly classify known categories but also to identify new categories that have not been seen before. Therefore, the common solution of OSFD is to first build a feature extraction model, then design a feature measurement method such as a distance-based method, and further set a reasonable threshold to evaluate whether the tested data belongs to a known class. Tian et al. proposed a feature fusion method based on $t$-sne and kernel null Foley–Sammon transform (KNFST) to construct the state feature space of the equipment, and then designed an adaptive threshold based on the test data to improve the accuracy of classifying known faults and identifying unknown faults.\textsuperscript{28} Lundgren and Jung employed a probability density function (PDF) to abstract the failure characteristics of the data and proposed a one-class classifier using the approximate distinguishability measure to test if a new PDF $p$ can be explained by a known fault mode $f_j$.\textsuperscript{29} To achieve OSFD, they used a one-class classifier for each fault mode $f_j$ and designed a tuning strategy for the threshold $K_j$ of each classifier to achieve the classification of known faults and the identification of unknown faults.

Different from OSFD, CDFD focuses on solving fault diagnosis problems across working conditions or across equipment. The main idea of solving CDFD is domain adaptation (DA),\textsuperscript{30,31} which is a technique of transfer learning.\textsuperscript{32} The core idea of DA is to reduce the distribution discrepancy between the source domain (SD) and the target domain (TD) to achieve the purpose of classifying the TD only by training the SD data or a small amount of TD data. Therefore, how to represent the features of the SD and the TD, how to measure the distribution difference between the SD and the TD, and how to reduce the distribution discrepancy between the SD and the TD have become the hot research directions of DA. Qin et al. adopted modified composite multi-scale fuzzy entropy (MCMFEs) as the fault features to express similar features information between the TD and the SD.\textsuperscript{33} Further, they proposed a feature similarity measurement method based on minimum redundancy maximum relevance (mRMR), Kullback–Liebler divergence (KLD), and k-nearest neighbor (KNN) to improve the classification accuracy of CDFD for rolling bearing. Li et al. present a knowledge mapping-based adversarial domain adaptation (KMADA) method to learn a distance metric measured by a discriminator.\textsuperscript{34} This method requires the construction of 2D feature images of each domain via short-time Fourier transform (STFT) and L2 normalization.

Both OSFD and CDFD are very complex in various stages of organizing data, building diagnostic models, and training diagnostic models. Because the models are too complex, the computational loss is also very large. Considering that in actual engineering, the occurring possibility of faults in the initial stage of equipment operation is relatively small. Before a device fails or new faults occur, we can often collect some valid data of the device. At the same time, for the equipment that fails, we usually shut down the equipment directly, and the technicians will conduct a detailed inspection to determine the specific fault mode and formulate maintenance strategies. Therefore, for an automatic diagnosis model, it is more engineering meaningful to be able to identify a new fault as an anomaly than to identify that the fault is a new fault.
In view of the abovementioned fault diagnosis engineering requirements, this article attempts to improve the generalization ability of simple networks, so that it can solve the problems of fault diagnosis and anomaly detection under the condition of multiple operating conditions and inconsistent class labels. This article mainly studies fault diagnosis and anomaly detection tasks under the condition that the category of testing data is larger than training data. It is found that Huffman coding can effectively reduce the noise of the vibration signal of the rotor equipment and enhance the impact characteristics of the vibration signal in our previous study. Considering that some effective feature information will be lost in the preprocessing of vibration signal, we try to combine Huffman coding with CNN, and conduct Huffman coding on the data stream during the CNN feature extraction process. In view of CNN’s weak ability to identify unknown test samples, we further enhanced CNN’s ability to describe normal data by increasing the distribution difference between classes of known training data to isolate the characteristic space of normal data. For this reason, we introduce mixed loss into the new network objective function. With the help of the Huffman coding and mixed loss, the network could improve the accuracy on classifying the testing data which has more diversity than the training data.

The main contributions of the proposed method are as follows:

1. We propose a new pooling method called Huffman pooling according to the principle of Huffman coding and introduce it into the traditional CNN. Compared with traditional pooling methods such as max pooling and average pooling, the Huffman pooling method can not only preserve the numerical information within each pooling unit but also the complexity information within the pooling unit. The Huffman pooling process effectively retains more information related to the original data features so it can effectively enhance the robustness of the CNN model.

2. We design a new H-CNN objective function based on softmax loss, intra-class loss, and inter-class loss. Compared with the traditional loss function, the new loss function can not only guide the model to learn the features of the same class of samples, but also guide the model to distinguish the features of different classes of samples so it can effectively improve the generalization ability of the model for fault diagnosis and anomaly detection under multiple operating conditions.

3. To verify the generalization ability of the H-CNN in fault diagnosis and anomaly detection under multiple working conditions, the proposed method is tested on three different datasets which are the cylindrical roller bearings dataset, deep groove ball bearings dataset, and gearbox dataset. In the detection tasks for known faults and unknown faults, the accuracy rate reaches 95.37/91.93, 98.62/95.27, and 99.79/96.32 respectively. Compared with other methods, the H-CNN has higher accuracy.

The rest of this article is organized as follows. The basic theory of Huffman average code length (HACL) and the CNNs is presented in Basic theory section. Proposed method for fault diagnosis section elaborates on the pooling method based on Huffman coding
theory, the structure of the fault diagnosis model based on Huffman pooling and CNN, and a new mixed loss function. In Case studies section, the identification performance of the proposed method is verified on three datasets. Finally, the concluding remarks are summarized in Conclusion section.

**Basic theory**

**HACL**

Huffman coding is a method of optimal prefix coding used for lossless data compression. Since David A. Huffman proposed the coding algorithm in 1952, Huffman coding has been widely used in the field of computer science and information theory due to its high efficiency. The output of Huffman’s algorithm is essentially a variable-length code table with which source symbols can be binary encoded. The algorithm derives the code table from the frequency of occurrence of all possible values of the source symbol.

Let the symbol set $A = \{a_1, a_2, \ldots, a_n\}$, and the corresponding probability is $P = \{p_1, p_2, \ldots, p_n\}$, $p_i = \text{probability}(a_i)$, $1 \leq i \leq n$. The Huffman code of $A$ is $C(P) = \{c_1, c_2, \ldots, c_n\}$, where $c_i$ is the codeword for $a_i$, $1 \leq i \leq n$. The HACL of $A$ can be solved as follows:

**Step 1:** Let $A^* = A$, $P^* = P$, $A^*$ and $P^*$ will change in the coding process. The size of $A^*$ and $P^*$ is $n$ at the beginning of the encoding.

**Step 2:** Let $A'$ be the set obtained by rearranging $A^*$ in descending order of probability $P^*$. $A' = \{a'_1, a'_2, \ldots, a'_{m-1}, a'_m\}$, $m$ is the size of $A$.

**Step 3:** Take the two symbols $a'_{m-1}$ and $a'_m$ with the least probability in the symbols sequence. Encode the symbol $a'_{m-1}$ with higher probability into “1” and the symbol $a'_m$ with less probability as “0”. Add the probabilities ($p'_{m-1}$ and $p'_m$) of the two symbols as the probability $p^*$ of the new symbol $a^*$. Delete $a'_{m-1}$ and $a'_m$ from $A'$. Add $a^*$ into $A'$. Then the set $A'$ has changed into $A''$ and the size of $A''$ is $m - 1$. Let $A^* = A''$.

**Step 4:** Repeat Step 2 and Step 3 until there is only one symbol in set $A^*$ and the probability of $a^*$ is 1.

**Step 5:** Backtrack from the symbol with the probability of 1 to each source symbol and record 0/1 in the backtracking path. The Huffman code of $a_i$ is $c_i$.

**Step 6:** The HACL of each source symbol $a_i$ is calculated according to the length of $c_i$ and the corresponding probability $p_i$ as follows:

$$HACL = \left( \sum_{i=1}^{n} p_i \right)^{-1} \sum_{i=1}^{n} p_i^2 \text{length}(c_i) \quad (1)$$

For a set of source symbols $U = \{u_1, u_2, u_3, u_4, u_5, u_6\}$ with probabilities $P = \{p_1, p_2, p_3, p_4, p_5, p_6\}$, $p_1 = 0.45$, $p_2 = 0.16$, $p_3 = 0.13$, $p_4 = 0.12$, $p_5 = 0.09$, and $p_6 = 0.05$, the process of Huffman coding is presented in Figure 1.

Let the code length of the $i$th symbol in $U$ is $l_i$. It can be seen from Figure 1 that $l_1 = 1$, $l_2 = 3$, $l_3 = 3$, $l_4 = 3$, $l_5 = 4$, and $l_6 = 4$. The HACL of the probability sequence is calculated as $HACL = p_1 * l_1 + p_2 * l_2 + p_3 * l_3 + p_4 * l_4 + p_5 * l_5 + p_6 * l_6 = 2.24$. 
CNNs

CNN is generally composed of three parts: input layer, hidden layer, and output layer. The input layer is the original data such as images. The output layer is the result of classifying the features. The hidden layers are neuron layers with complex multilayer nonlinear structure, including convolution layers, activate layers, subsampling layers, and perceptron layers. A convolution layer, an activate layer, and a subsampling layer are combined into a feature extraction unit. The perceptron layers classify features by mapping them to class labels. The classification ability of the CNN model can be improved by optimizing the feature extraction layers and the perceptron layers. The structure of the CNN is shown in Figure 2.

Figure 2 shows a simple CNN with two convolutional layers (C1, C2), two activate layers, and two subsampling layers (S1 and S2) in the feature extraction layer. After multiple feature extractions, the single-layer perceptron transforms the final 2D feature map into 1D matrix. The output layer divides the features into N classes. The single-layer perceptron is usually a fully connected structure.

Convolutional layer. Convolutional layers are the core component of hidden layers. A convolutional layer is composed of multiple stacked convolution kernels. The convolution
kernel is like a filter, which can extract features at a smaller scale on the data input to the convolutional layer. The information of the input data is enhanced, and the interference of noise is suppressed. The size of the convolution kernel is called the receptive field. The input and output of a convolutional layer are both feature maps. The value of the output feature map can be calculated by equation (2).

\[ z_k = w_k \otimes x + b_k \tag{2} \]

where \( w_k \) is the weight vector of the \( k \)th convolution kernel, \( x \) is the input feature map, \( b_k \) is the bias of the \( k \)th convolutional kernel, \( z_k \) is the output feature map, and \( \otimes \) is convolution operation. For example, there is an input feature map \( x \) with a size of \( 4 \times 4 \), a convolution kernel \( w_k \) with a size of \( 2 \times 2 \), and the sliding step parameter \( stride \) of the convolution kernel is 1. The process of the convolution operation by \( k \)th convolution kernel can be expressed as shown in Figure 3.

Activate layer. CNN simulates the response of human brain nerves to different stimuli using nonlinear activation layers. The introduction of nonlinear activation layers enables CNNs to handle complex computational problems. Commonly used activation functions include sigmoid function, Tanh function, and ReLU function. The formulas of these functions are as follows:

\[ \text{Sigmoid}(x) = \frac{1}{1 + e^x} \tag{3} \]

\[ \text{Tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{4} \]

\[ \text{ReLU}(x) = \max(0, x) \tag{5} \]

Subsampling layer. The subsampling layer reduces the amount of computation and abstracts the features by downsampling the output feature map of the convolution layer. Subsampling layers are also known as pooling layers. Commonly used pooling methods are max pooling and average pooling. For example, the size of a feature map processed by an activation layer is \( 4 \times 4 \), the pooling size is \( 2 \times 2 \), and the sliding step parameter \( stride \) is 2. The common pooling process is shown in Figure 4.
Perceptron layer. The perception layer is usually a fully connected structure. The fully connected layers are the last few layers in the CNN and act as the classifier. Another main role of the perception layer connected to the hidden layer is to convert the 2D feature map into a 1D feature matrix. Each node of the fully connected layer is fully interconnected with the nodes of the previous layer. The fully connected layer performs weighted summation of the features output by the previous layer and inputs the result to the output layer. The formula of the fully connected layer is as follows:

\[ z_i = \sum_{i=1}^{n} w_i x_i + b_i \]  \hspace{1cm} (6)

where \( w_i \) is the weight coefficient of the fully connected layer, \( x_i \) is the value of the \( i \)th neuron in the previous layer, and \( b_i \) is the bias of the fully connected layer.

Output layer. For classification issues, the softmax function is usually applied in the output layer. The formula of the softmax function is as follows:

\[ \text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{c=1}^{C} e^{z_c}} \]  \hspace{1cm} (7)

Proposed method for fault diagnosis

CNN updates the weights of the network by the back-propagation algorithm based on the loss between the predicted label and the true label. The network could obtain a high identification accuracy once the loss function is converged to a global minimum value. Considering that the working conditions of rotating equipment in engineering practice are complex and changeable, this section proposes a new loss function and a new network structure to improve the generalization ability of CNN.

Generally, the objective function of CNN is defined to minimize the loss between the predicted table and the real table. When the loss function converges to the global minimum, CNN has a high recognition accuracy. Considering the complex and changeable working conditions of rotating equipment in engineering practice, the data categories to be detected are far more than the available data categories, so CNN cannot learn all the
labels to be tested. Therefore, a new loss function and a new network structure are proposed in this section to improve the generalization ability of CNN and try to solve the problem of fault diagnosis and anomaly detection under the condition of insufficient categories of training data.

**Signal to 2D image conversion method**

Since most of the traditional data-driven methods for diagnosis cannot directly process the original data, many researchers design different data preprocessing methods to improve the feature expression ability of the original data. However, the characteristics obtained by different treatment methods are also different, so it has a great influence on the result. Wen et al. proposed an effective data preprocessing method to maximize the retention of the original data.\(^{25}\) The idea of this method is to convert the time-domain signal into an image by fulfilling the pixels calculated by the value of the signal. For a signal \(L(n)\), the image can be obtained as follows.

**Step 1:** Select a segment \(L'(M^2)\) with length \(M^2\) randomly from \(L(n)\), \(M^2 \leq n\), \(L'(M^2) = \{L'_1, L'_2, \cdots, L'_{M^2}\}\).

**Step 2:** Generate a blank image of size \(M \times M\).

**Step 3:** Let \(M(i, j)\) be the pixel strength of the image, \(i = 1, 2, \cdots, M\), \(j = 1, 2, \cdots, M\). The pixels of the image can be fulfilled by equation (8), where \(\text{round}(\ast)\) is the rounding function.

\[
M(i, j) = \text{round}\left\{\frac{L'_((i-1)M+j) - \text{Min}(L')}{\text{Max}(L') - \text{Min}(L')} \times 255\right\} \tag{8}
\]

Figure 5 shows the image converted from raw signals.

**Huffman pooling**

It can be seen from the above Huffman coding process that if Huffman coding is performed on a time series, the amount of data can be effectively reduced, and the dimensionality reduction effect can be achieved.

**Figure 5.** Signal to 2D image conversion.
To analyze the influence of Huffman coding on the data better, this article takes the famous dataset of bearing vibration signal of the famous Case Western Reserve University as an example. Since the vibration signal is a series of vibration amplitudes in the time domain, the data itself usually contains negative values. According to the method of solving the HACL, the signal needs to be converted into a discrete point sequence with probability meaning. Therefore, the original signal needs to be processed into a nonnegative sequence first and then converted into a probability sequence. As shown in Figure 6, the probability sequence is transformed into HACLs finally.

For a signal data $X(n) = \{x_1, x_2, \cdots, x_n\}$, the HACL sequence is obtained as follows:

**Step 1:** According to equation (9), convert signal amplitude to nonnegative value $X'(n) = \{x'_1, x'_2, \cdots, x'_n\}$.

$$x'_i = x_i - \min[X(n)], \quad i = 1, 2, \cdots, n \quad (9)$$

**Step 2:** According to equation (10), transform $X'(n) = \{x'_1, x'_2, \cdots, x'_n\}$ to probability sequence $P(n) = \{p_1, p_2, \cdots, p_n\}$.

$$p_i = \frac{x'_i}{\sum_{j=1}^n x'_j} \quad (10)$$

**Step 3:** Divide $P(n) = \{p_1, p_2, \cdots, p_n\}$ into different segments $G(m) = \{g_1, g_2, \cdots, g_m\}$ with the sliding window, where $g_i = \{p_{i+s(i-1)}, p_{i+s(i-1)+1}, \cdots, p_{i+s(i-1)+l-1}\}$, $l$ is the length of the window, $s$ is the step size of sliding window movement. The size of $G(m)$ can be solved according to equation (11), where $\text{floor}(\ast)$ means finding the biggest integer

![Figure 6. Probability sequence-to-HACLs conversion method.](image)

HACL: Huffman average code length.
smaller than \( x \).

\[
m = \text{floor} \left( \frac{n - l}{s} \right) + 1
\]

(11)

**Step 4:** Calculate the HACL sequence as

\[
\text{HACLs} = \{ \text{HACL}(g_1), \text{HACL}(g_2), \cdots, \text{HACL}(g_m) \}.
\]

This article applies the HACLs method to the SKF 6205-2rs bearing test data of Western Reserve University.\(^{37}\) The vibration signal was collected under 0hp load condition, and the corresponding speed was 1797 r/min. There are four types of data in this dataset, including normal (N), inner race fault (IF), outer race fault (OF), and ball fault (BF). Each type of fault contains three damage sizes (0.178 mm, 0.356 mm, and 0.533 mm). The HACLs for the above 10 different types of vibration signals are calculated by setting \( l = 5, 8, 11, 16, 20, 25 \) and \( s = 1, 4, 7, 10 \).

To better illustrate the influence of HACLs on the signal, this article analyzes from three perspectives of time domain, frequency domain, and distribution in 2D feature space.

The waveforms of the raw signals and their HACLs with \((l, s) = (5, 1)\) are shown in Figure 7. It can be seen that the HACLs can weaken the common vibration component in the original signal to a certain extent in the time domain. The impact component of the signal is enhanced at the same time.

Combined with the relevant parameters used in the bearing experiment, it can be calculated that the rotating frequency is 29.95Hz under the 0hp load condition. The characteristic frequencies corresponding to IF, OF, and BF are 162.19Hz, 107.36Hz, and

![Figure 7](image-url)  
**Figure 7.** The waveforms of raw signals and their HACLs with \((l, s) = (5, 1)\). (a) Normal (N), (b) outer race fault (OF), (c) inner race fault (IF), and (d) ball fault (BF). HACLs: Huffman average code length.
141.17Hz respectively. For the raw signals as shown in Figure 7, the amplitude spectrum and envelope spectrum were analyzed respectively. For the HACLs, only the amplitude spectrum was analyzed. The results are presented in Figure 8.

It can be seen from Figure 8 that it is difficult to obtain the characteristic frequency of the original signal in the low-frequency band by directly analyzing the amplitude spectrum. When Fourier spectrum analysis of the original signal is performed directly, the fault frequency will exist in the sideband of the high frequency, which makes it difficult to intuitively find the fault frequency in the low-frequency band. However, by calculating the envelope spectrum of the original signal, it can be found that the characteristic frequency can be observed very intuitively in the low-frequency band. By comparing the envelope spectrum for raw signals and amplitude spectrum for HACLs, it is not difficult to find that for HACLs, only amplitude spectrum analysis is needed to get the obvious fault characteristic frequency in the low-frequency band. It indicates that HACLs has a similar effect to the envelope spectrum.

To better illustrate the influence of HACLs on the distribution in 2D feature space, the t-sne method was applied to visualize the original data and their HACLs with different parameters.38 The feature distribution in 2D space is presented in Figure 9.

It is obvious that the data of the same type and degree of failure in the original data are not concentrated in the feature space, but HACLs with appropriate parameters in the feature space show more concentration of the same type of data, and low concentration

![Figure 8. The frequency analysis results of raw signals and their HACLs. (a) Normal (N), (b) outer race fault (OF), (c) inner race fault (IF), and (d) ball fault (BF). HACLs: Huffman average code length.](image)
Figure 9. The feature distribution of raw signal and HACLs in 2D space. (a) Raw signal and (b) HACLs with different parameters. The “-1”, “-2”, and “-3” in the legend means the damage size of the fault is 0.178 mm, 0.356 mm, 0.533 mm respectively.

HACLs: Huffman average code length.
of different types of data. In addition, the distribution of HACLs in the feature space also has the following characteristics:

1. When the parameters are selected properly, HACLs can increase the degree of dispersion between normal data and abnormal data, which has guiding significance for quickly determining whether the rotor equipment is abnormal in engineering practice;

2. No matter what parameters are selected, the distribution area of the rolling element failure samples is always close to the areas of the other three types of samples, which is consistent with the propagation mechanism of rolling element failures. Due to the particularity of the rolling element structure, when it fails, it will indirectly cause the abnormal vibration of the inner ring, the outer ring, and the entire experimental equipment by means of vibration transmission, and as the effect of the fault continues to spread, the failure energy of the rolling element itself will gradually decrease, so whether in the time-frequency domain analysis or in the characteristic distribution analysis, the fault characteristics of the rolling element itself are easily covered by the fault characteristics of the inner and outer rings;

3. It can be seen from Figure 9 that when $l$ is constant and $s$ has less influence on the distribution of HACLs in the feature space. But when $s$ is constant and $l$ changes from small to large, the concentration of HACLs of the same data varies greatly. For example, when $l = 11, 20, 25$, the degree of concentration is high, but when $l = 8, 16$, the degree of concentration is low.

From the above analysis of the distribution of HACLs in the time domain, frequency domain, and feature space, the following conclusions can be drawn: HACLs can effectively enhance the impact components in the original signal, remove high-frequency components, and keep the fault characteristic frequency unchanged. At the same time, HACLs can enhance the degree of aggregation of the same type of original data in the feature space and provide convenience for further analysis of the health status and fault type of the equipment. However, different calculation parameters have a greater impact on the distribution characteristics of the data in the feature space.

In view of the characteristics of the abovementioned HACLs, this article proposes a method of fusing HACLs and CNN. By applying HACLs to the pooling operation of CNN, it can not only reduce the dimensionality in the pooling process, but also retain more features of the original signal.

According to the above introduction to the Huffman coding method, the process of solving HACL fully considers the impact of all data. Therefore, this method can be applied to the pooling process of the CNN.

CNN is a network connected layer by layer with three different functions: convolutional layer; pooling layer; fully connected layer. The convolutional layer applies multiple filters to obtain the feature map of the input image. The pooling layer abstracts the features by reducing the input feature dimension by means of downsampling. The features are extracted by alternating convolutional layer and pooling layer. The abstract features are finally input to the full connection layer to calculate the classification results.
Set the output size of the \(i\)th convolutional layer is \(N \times M\), the \(i\)th Huffman pooling kernel size is \(n \times m\). The sliding steps of the pooling kernel along the rows and columns are \(\text{step}_n\) and \(\text{step}_m\) respectively. These parameters need to satisfy the following constraints: (i) \(N-n\) is divisible by \(\text{step}_n\) and (ii) \(M-m\) is divisible by \(\text{step}_m\). Figure 10 shows how the Huffman pooling layer works.

According to the method of calculating HACL, the data needs to be converted into a discrete point sequence with probability meaning. Therefore, the output data of \(i\)th convolutional layer needs to be processed into a nonnegative data first and then converted into a probability sequence. As shown in Figure 10, the output of \(i\)th convolutional layer is normalized before pooling.

For a 2D data \(X(N, M) = \{x_{11}, x_{12}, \ldots, x_{1M}; x_{21}, x_{22}, \ldots, x_{2M}; \ldots; x_{N1}, x_{N2}, \ldots, x_{NM}\}\), the HACL 2D data is obtained as follows:

**Step 1:** According to equation (12), transform the signal amplitude to nonnegative value \(X'(N, M) = \{x'_{11}, x'_{12}, \ldots, x'_{1M}; x'_{21}, x'_{22}, \ldots, x'_{2M}; \ldots; x'_{N1}, x'_{N2}, \ldots, x'_{NM}\}\).

\[
x'_{ij} = x_{ij} - \min\{X(N, M)\}, \quad i = 1, 2, \ldots, N; \quad j = 1, 2, \ldots, M
\]  

**Step 2:** According to equation (13), transform \(X'(N, M)\) to \(P(N, M)\),

\[
P(N, M) = \{p_{11}, p_{12}, \ldots, p_{1M}; p_{21}, p_{22}, \ldots, p_{2M}; \ldots; p_{N1}, p_{N2}, \ldots, p_{NM}\}, \quad p_{ij} \in [0, 1].
\]

\[
p_{ij} = \frac{x'_{ij}}{\sum_{i=1}^{N} \sum_{j=1}^{M} x'_{ij}}
\]  

**Step 3:** Divide \(P(N, M)\) into different blocks according to the sliding window with step size \(\text{step}_n\) and \(\text{step}_m\).

**Step 4:** Calculate each block as HACL into \(G(N', M')\). The size of \(G(N', M')\) can be calculated according to equation (14).

\[
N' = \frac{N - n}{\text{step}_n}, \quad M' = \frac{M - m}{\text{step}_m}
\]
This article compares the differences between Huffman pooling and the traditional pooling methods with three sets of examples. The three sets and the results of different methods are listed in Table 1. Obviously, when the data length and shape change, the results calculated by the max pooling and average pooling methods may be the same, but the results calculated by the Huffman pooling method are significantly different. Therefore, the Huffman pooling method can effectively reduce the data dimension while retaining the differences between different data.

It can be seen from Table 1 that for three different sets of data, the results of max pooling and average pooling methods are the same, which cannot reflect the difference of the original data, but the results of Huffman pooling on different data are obviously different. Therefore, the Huffman pooling presented in this article can not only achieve dimensionality reduction sampling, but also retain the different characteristics between the original data.

**Huffman-CNN (H-CNN) construction**

The ability of CNN to extract and classify features of research objects has been widely verified in the field of image recognition. For signals that have been converted to images, a CNN can train them and classify them well.

This article proposes the H-CNN based on the Huffman pooling and CNN mentioned above. The network structure includes a feature extraction stage and a classification stage. The feature extraction contains convolution layers, batch normalization layers, ReLU activation layers, and pooling layers. The classification stage is realized by the full connection layer.

The structures of the typical CNN and H-CNN for $60 \times 60$ images are shown in Figure 11. In these two structures, we set $N = M$ and $step_n = step_m = n = m$. The parameters on each layer are presented in Table 2. The denotation of Conv ($5 \times 5 \times 32$) means that it is a convolutional layer where the filter size is $5 \times 5$ with 32 channels. Maxpool ($2 \times 2$) means that it is a maxpool layer with ($2 \times 2$) filters. Huffman pool ($2 \times 2$) means that it is a Huffman pooling layer with ($2 \times 2$) filers.

**The new mixed loss function**

Minimizing the distance between the distribution of training set and testing set is a common method to reduce generalization error on test set. When there are unknown faults in the diagnosis task, it is very difficult to predict the distribution of test set directly.
because no prior information about fault mode and fault degree can be obtained. To solve the above problems, a feasible solution is to make full use of the training set and establish the feature boundary between normal samples and fault samples by CNN. Although the feature boundary cannot be directly used to diagnose unknown fault types, it can be used to determine whether unknown data is abnormal. Therefore, a new mixing loss function is designed to improve the ability of the diagnostic model in homogeneous aggregation and heterogeneous differentiation. This is also an attempt to construct the feature boundary.

Generally, the loss function of CNN is designed according to the difference between the output of the network and the real label. As the training set of $\{(X_i, y_i)\}_{i=1}^n$, $X_i$ is the input data, $y_i$ is the corresponding label, $n$ is the number of classes. The corresponding

---

**Figure 11.** The structure of the neural networks: (a) A typical convolutional neural network (CNN) structure for $60 \times 60$ images and (b) a Huffman-convolutional neural network (H-CNN) structure for $60 \times 60$ images.
output of CNN is $y'_i$. The traditional loss function can be represented as follows:

$$L = \sum_{i=1}^{n} |y'_i - y_i|$$  \hspace{1cm} (15)$$

In practice, a softmax classifier is usually used to classify the output of CNN, so the loss function is usually designed as follows:

$$L_s = \min_{y} \left(-\sum_{i=1}^{m} \log \sum_{j=1}^{p} e^{W_j^* x_i + b_j} \right)$$ \hspace{1cm} (16)$$

where $m$ is the size of mini batch. $x_i$ is the $i$-th feature belonging to the $y_i$. $W_j$ and $b_j$ is the $j$-th column of the weight and bias of the last fully connected layer, respectively.

In this article, we introduce a mixture loss function considering homogenous clustering and heterogeneous differentiation in traditional CNN. By minimizing intra-class variation and maximizing inter-class variation, the new loss can distinguish the characteristics of normal data and fault data. Under the guidance of mixed loss and softmax loss, the network can explore some commonalities of feature spaces under different working conditions. By learning the characteristics of known health data, the ability to identify unknown data as abnormal data can be improved. In this way, the network has good generalization performance for unknown health data.

The mixed loss function is proposed with reference to the idea of center loss. In the DL method for face recognition,\(^{39}\) the center loss is designed to enhance its recognition ability. The main idea behind center loss is to simultaneously learn the center of each class feature and penalize the distance between the feature and the corresponding class center.

In order to minimize the feature distance of intra-class and maximize the feature distance of inter-class, this article proposes a mixed loss function based on the distance between center point and feature. For the given set $\{(X_i, y_i)\}_{i=1}^{n}$. The mixed loss is
designed as follows:

\[
L_M = \min \left( \frac{1}{2} \sum_{i=1}^{n} \|X_i - c_{y_i}\|^2 + \frac{1}{\sqrt{\frac{1}{2} \sum_{i=1}^{n} |c_{y_i} - c_{y_j}|^2}} \right), \quad (i \neq j)
\]  

(17)

where \(c_{y_i}\) is the center point of features of class \(y_i\). In theory, \(c_{y_i}\) should be computed considering the entire training samples. However, the training process of CNN relies on the minimum batch of samples. The center point can be updated by replacing the entire training set with small batches. In each iteration, the update of \(c_{y_i}\) can be processed as equations (18) to (19). In this article, the scalar parameter \(\alpha\) is set as the default value 0.5.

\[
\Delta c_{y_i} = \sum_{i=1}^{n} \delta(y_i = k)(c_k - X_i) \\
1 + \sum_{i=1}^{n} \delta(y_i = k), \quad \delta = \begin{cases} 
1 & y_i = k \\
0 & y_i \neq k
\end{cases}
\]

(18)

\[
c_{y_i}^{t+1} = c_{y_i}^t + \alpha \Delta c_{y_i}^t
\]

(19)

To ensure the features of clustering and dispersion between classes, a new mixed loss is added based on softmax loss. Finally, the new CNN is trained by optimization algorithm to minimize the loss function. The final loss function of H-CNN is defined as follows:

\[
L_{H-CNN} = L_s + \mu L_M
\]

(20)

where \(\mu\) is a scalar parameter, we set \(\mu = 1e^{-6}\) considering the order of magnitude difference between \(L_s\) and \(L_M\).

**Case studies**

This article focuses on the bearings and gears which are most widely used as rotor components in mechanical equipment for fault diagnosis and abnormal detection based on vibration signal drive.

In this section, the proposed method H-CNN is conducted on three different datasets, which are the cylindrical roller bearings dataset, deep groove ball bearings dataset, and gearbox dataset. The comparison test is mainly for the H-CNN and some other methods.

**Data acquisition**

Many types of faults and failure modes can occur to bearings and gears in various rotating machinery systems. Vibration signals collected from such a system are usually used to reveal its health condition. There are three datasets to verify the proposed method, including two bearing datasets and a gearbox dataset.
In two bearing datasets, experimental data were collected on a test rig with symmetrical structure and replaceable bearings as shown in Figure 12. The experimental device is mainly composed of a driving motor, a loading system, a shaft, a conveyor belt, and two bearings. The fault parts are prepared by means of electro-discharge machining. The bearing data used in this article are all vibration signals in the vertical direction.

In the gearbox dataset, the time-domain fault vibration data of gear is downloaded from the FigShare website. Cao et al. completed the experiment. The vibration signals are collected from a two-stage gearbox that is driven by a motor and loaded through a magnetic brake as shown in Figure 13. The torque is provided by a magnetic brake, which can be adjusted by changing the input voltage. A 32-tooth pinion and an 80-tooth pinion are mounted on the first stage input shaft. The second stage consists of a 48-tooth pinion and a 64-tooth pinion. The speed of the gears is controlled by an electric motor. Input shaft speed is measured by tachometer and gear vibration signals are collected by an accelerometer.

**Case 1: Cylindrical roller bearing.** In the first bearing dataset, the test bearing is N205, a cylindrical roller bearing. The sampling frequency is 20 KHz. Seven health conditions are set as: normal; inner race with wear fault; outer race with breakage fault; roller with pitting fault; roller with groove fault; roller with wear failure; roller with missing piece fault. Figure 14 shows the fault bearings. Cylindrical roller bearings were tested under 15 load conditions, with the speed ranging from 100 r/min to 1500 r/min. For each health condition under a single load condition, the vibration signals were processed into a series of samples with a length of 3600.

The samples are divided into training set and testing set. In order to prove that the method proposed in this article can identify whether the data of unknown health state is normal or not, we only use the normal data and four fault data, as shown in Figure 14(a) to (d) to train the network. The description of the cylindrical roller bearing data samples is listed in Table 3.

![Figure 12. The bearing experimental rig.](image-url)
Case 2: Deep groove ball bearing. In the second bearing dataset, the test bearing is 6312, a deep groove ball bearing. The sampling frequency is 10 KHz. There are four load conditions, and the corresponding speeds range from 300 r/min to 1500 r/min (300/700/1100/1500). Five health conditions are set as normal; inner race with breakage fault; outer race with breakage fault; retainer with breakage fault; and ball with wear failure. The fault bearings are shown in Figure 15. For each health condition under a single load condition, the vibration signals were processed into a series of samples with a length of 3600. The samples are also divided into training set and testing set. We use the normal data and three fault data, as shown in Figure 15(a) to (c) to train the network. A detailed description of the samples is listed in Table 4.

Case 3: Gearbox system. In the gearbox dataset, the sampling frequency is 20 KHz. The nine different conditions are introduced to the pinion on the input shaft: normal; missing
tooth; root crack; spalling; and chipping tip with five different levels of severity ("chip5a", "chip4a", "chip3a", "chip2a", and "chip1a"). Figure 16 shows the different health conditions. For each health condition, the vibration signals are sliced into a series of samples with a length of 3600.

Figure 14. Six fault conditions of cylindrical roller bearings: (a) Inner race with wear fault, (b) outer race with breakage fault, (c) roller with pitting fault, (d) roller with groove fault, (e) roller with wear failure, and (f) roller with missing piece fault.
The samples are divided into training set and testing set. We use the normal data and six fault data (missing tooth/root crack/spalling/chip5a/chip4a/chip3a) to train the network. A detailed description of the samples is listed in Table 5.

**Comparison methods**

To evaluate the performance of the proposed H-CNN model, traditional CNN and some other methods are used in this article to compare the prediction accuracy of the three cases.

The traditional CNN methods used for comparison in this article include CNN-2D and CNN-1D. CNN-2D uses the same input data as H-CNN. Compared to H-CNN, CNN-2D only differs in the pooling layer. The structure of CNN-2D is shown in Figure 11(a) and Table 2. The input of CNN-1D is a 1D raw signal that has not been converted into 2D image. Meanwhile, the handcrafted features are extracted in time domain and frequency domain. The time domain is characterized by peak, peak to peak, absolute mean, standard variance, margin index, peak index, impulse factor, skewness index, and kurtosis index. In the frequency domain, there are six features, which are frequency center, root mean

---

**Table 4.** Label description of the deep groove ball bearing dataset.

| Class label | Load speed (r/min) | Health condition          | Training set | Testing set |
|-------------|--------------------|----------------------------|--------------|-------------|
| 1           | 300/700/1100/1500  | Normal                     | 640          | 160         |
| 2           | 300/700/1100/1500  | Inner race with breakage fault | 640          | 160         |
| 3           | 300/700/1100/1500  | Outer race with breakage fault | 640          | 160         |
| 4           | 300/700/1100/1500  | Retainer with breakage fault | 640          | 160         |
| 5           | 300/700/1100/1500  | Ball with wear failure     | –            | 160         |

---

**Figure 15.** Four fault conditions of deep groove ball bearings: (a) Inner race with breakage fault, (b) outer race with breakage fault, (c) retainer with breakage fault, and (d) ball with wear failure.
square frequency, SD frequency, and CP1, CP2, and CP3. These statistical features are applied to shallow methods to classify health status. In this article, random forest (RF), SVM, KNN, and artificial neural network (ANN) are selected. In addition, some other advanced DL methods, including DBN, LSTM, and ResNet, are also used to compare and verify the effectiveness of the proposed method.

The average results of 10 tests are used as the evaluation basis in this article to reduce the error caused by randomness.

**Results and discussion**

**Case 1: Cylindrical roller bearing.** The images (60 × 60) converted from cylindrical roller bearing raw signals are shown in Figure 17. It can be observed that the images under different operating conditions and different health conditions are totally different.
Figure 17. N205 cylindrical roller bearing raw signals conversion results: (a) Inner race with wear fault, (b) outer race with breakage fault, (c) roller with pitting fault, (d) roller with groove fault, (e) roller with wear failure, and (f) roller with missing piece fault.
As shown in Table 3, the training data contains five classes of data and the testing data contains seven classes of data in case 1. The accuracy of the recognition, in this case, is defined as follows:

\[
\text{Accuracy} = \begin{cases} 
\frac{N_{\text{predict}}(\text{Label}_{\text{predict}} = \text{Label}_{\text{real}})}{N_{\text{label}}^{1-5}} & \text{Label: } 1-5 \\
\frac{N_{\text{predict}}(\text{Label}_{\text{predict}} \neq \text{Label}_{\text{real}})}{N_{\text{label}}^{6-7}} & \text{Label: } 6-7
\end{cases}
\]

(21)

where \(\text{Label}_{\text{predict}}\) is the output of the test data, \(\text{Label}_{\text{real}}\) is the real label, and \(N_{\text{predict}}(\text{Label}_{\text{predict}} = \text{Label}_{\text{real}})\) is the number of output labels is the same as the number of real labels. For the fault test data shown in Figure 17(e) and (f), as long as the network recognizes that the data is not normal, we assume that the prediction is correct.

The results of case 1 are shown in Table 6. It is obvious that the proposed H-CNN method obtains a better result compared with other methods in this case. The average prediction accuracy is as high as 95.37% for known data and 91.93% for unknown data. It is obvious that the proposed H-CNN performs better to deal with the unknown data. Although the traditional CNN-2D achieves higher accuracy for the known data, it is weak in identifying the classes which do not exist in the training data.

The confusion matrices of different methods in case 1 are shown in Figure 18. SVM, CNN, and H-CNN mistake the data of fault (roller with wear failure) as normal data probability is 35.74%, 13.37%, and 9.65%, respectively. As for the fault (roller with missing piece fault), the misjudgment rates are 23.58%, 9.31%, and 6.49%, respectively. It is obvious that the traditional methods have poor ability to distinguish class labels 6–7. One of the possible reasons is that the proposed loss function can effectively enhance the difference between normal and abnormal data extracted by the network. So, the H-CNN has a better ability to identify fault data when the diversity of testing set is more than the training set.

Compared with case 2 and case 3, case 1 has more training data and test data, so case 1 is taken as an example to analyze the fault diagnosis performance of the proposed model.
under cross-working conditions. The dataset in Case 1 is reorganized into domains as shown in Table 7. The CDFD tasks are shown in Table 8.

We compared the proposed method H-CNN with CNN-2D and CNN-1D on the above seven CDFD tasks. The comparison results are shown in Figure 19. In Figure 19, H-CNN performs better than CNN-2D to classify known faults on tasks T2, T3, T4, and T6. For identifying unknown fault, H-CNN outperforms CNN-2D and CNN-1D in all seven tasks. At the same time, compared with CNN-2D and CNN-1D, H-CNN has smaller accuracy deviation, more stable performance, and higher reliability of diagnostic results.

According to the trends shown in Figure 19, we grouped and analyzed the seven tasks according to the load speed of the SDs and TDs.

(a) T2/T3/T5–T4/T6/T7: The average accuracy of T2/T3/T5 are lower than T4/T6/T7. In T2/T3/T5 tasks, the load speeds of the SDs are slower than that of the TDs. T4/T6/T7 tasks are just the opposite.

### Table 6. Comparison result of Huffman-convolutional neural network (H-CNN) and other methods in case 1 (%).

| Methods      | Average accuracy |
|--------------|------------------|
| H-CNN        | 95.37/91.93      |
| CNN-2D       | 97.79/88.66      |
| CNN-1D       | 92.54/78.32      |
| DBN          | 88.36/73.24      |
| LSTM         | 91.62/65.57      |
| ResNet       | 92.85/74.25      |
| RF           | 81.64/71.46      |
| KNN          | 72.39/65.83      |
| SVM          | 77.63/70.34      |
| ANN          | 68.43/59.63      |

ANN: artificial neural network; CNN: convolutional neural network; DBN: deep belief network; H-CNN: Huffman-convolutional neural network; KNN: k-nearest neighbor; RF: random forest; SVM: support vector machine.

### Table 7. Information of each domain in case 1.

| Domains | Load speed (×100 r/min) | Categories (labels) | Number of samples |
|---------|-------------------------|---------------------|-------------------|
| A       | 1/3/5/7/9/11/13/15      | 1/2/3/4/5          | 8 × 5 × 100 = 4000 |
| B       | 2/4/6/8/10/12/14        | 1/2/3/4/5/6/7      | 7 × 7 × 100 = 4900 |
| C       | 1/2/3/4/5               | 1/2/3/4/5          | 5 × 5 × 100 = 2500 |
| D       | 6/7/8/9/10              | 1/2/3/4/5          | 5 × 5 × 100 = 2500 |
| E       | 11/12/13/14/15          | 1/2/3/4/5          | 5 × 5 × 100 = 2500 |
| F       | 1/2/3/4/5               | 1/2/3/4/5/6/7      | 5 × 7 × 100 = 3500 |
| G       | 6/7/8/9/10              | 1/2/3/4/5/6/7      | 5 × 7 × 100 = 3500 |
| H       | 11/12/13/14/15          | 1/2/3/4/5/6/7      | 5 × 7 × 100 = 3500 |
(b) T2/T3–T6/T7: The average accuracy of T3 is lower than T2 while the accuracy of T7 is higher than T6. When the load speed of the SD is lower than that of the TD, the greater the difference between the speeds, the lower the accuracy. Conversely, the greater the difference between the speeds, the higher the accuracy.

(c) T1–T2/T3/T4/T5/T6/T7: Compared with other tasks, T1 has the smallest difference between the working conditions of the TD and the SD, so the accuracy of T1 is higher than that of other tasks, which is in line with common sense.

Based on the above analysis, we guess that the fault features under high-speed conditions are more compatible than those under low-speed conditions. To further verify this
conjecture, we independently trained and tested the data of each working condition. The results are shown in Figure 20. It is obvious that in the cylindrical roller bearing data set, the classification accuracy of low-speed conditions is always less than that of high-speed conditions. Under the test conditions of 100 r/min and 200 r/min, the classification accuracy of the known dataset is 86.39% and 90.47%, respectively. But it achieves to about 99% under the higher speed conditions. One possible reason is that the sampling frequency (20 kHz) is too high, while the data length of the sample (3600) is too short. When the time domain signal is converted to the angle domain signal and the operating condition is 100 r/min, the rolling bearing rotates only 0.3 turns. Considering the fault frequency of rolling bearings, the fault characteristics are not obvious under the limited data length. It is difficult to extract useful features at low speeds. But when the proposed method H-CNN is utilized, the classification accuracy is improved under both low-speed conditions and high-speed conditions. This shows that the method proposed in this article can effectively enhance CNN’s perception of less fault information.

It can be seen from the calculation method of Huffman pooling that Huffman pooling fully considers the impact of each value in the filter. In order to intuitively show that Huffman pooling can weaken the common components of signals and strengthen the impact components, we calculate the HACL directly on the original signal and then transform the signal into 2D image. As shown in Figure 21, a raw signal is converted into 2D image by two different methods. It can be seen that HACL has a good capability of noise reduction. So the Huffman pooling method can effectively improve the noise reduction ability of CNN and the sensitivity of CNN to the shock component in the signal.

**Case 2: Deep groove ball bearing.** The images (60 × 60) converted from deep groove ball bearing raw signals are shown in Figure 22.
As shown in Table 4, the training data contains four classes of data and the testing data contains five classes of data. The accuracy of the recognition is defined as follows:

$$\text{Accuracy} = \begin{cases} 
\frac{N_{\text{predict}}(\text{Label}_{\text{predict}} = \text{Label}_{\text{real}})}{N_{\text{label}:1-4}} & \text{Label}:1 - 4 \\
\frac{N_{\text{predict}}(\text{Label}_{\text{predict}} \neq \text{Label}_{\text{real}})}{N_{\text{label}:5}} & \text{Label}:5 
\end{cases}$$

(22)

The results of case 2 are shown in Table 9.

Figure 21. A raw signal converted into 2D image by different methods. CNN: convolutional neural network; H-CNN: Huffman-convolutional neural network.

Figure 22. 6312 deep groove ball roller bearing raw signals conversion results: (a) Inner race with breakage fault, (b) outer race with breakage fault, (c) retainer with breakage fault, and (d) ball with wear failure.
It can be seen that the proposed H-CNN method obtains a better result compared with other methods in this case. The mean prediction accuracy is as high as 98.62% for known data and 95.27% for unknown data. For the known data, the prediction results of CNN-2D, CNN-1D, DBN, LSTM, ResNet, RF, KNN, SVM, and ANN are 98.35%, 90.74%, 85.48%, 93.24%, 91.82%, 91.82%, 93.24%, 85.48%, and 93.24%, respectively. As for unknown data, the prediction results of CNN-2D, CNN-1D, DBN, LSTM, ResNet, RF, KNN, SVM, and ANN are 91.58%, 96.87%, 97.35%, 97.35%, 97.35%, 97.35%, 97.35%, 97.35%, and 97.35%, respectively. Table 9 shows that the proposed H-CNN has a good capability in fault diagnosis. Since there are fewer fault types in this case than in Case 1, the recognition accuracy, in this case, is generally higher than that in case 1.

The confusion matrices of different methods in case 2 are shown in Figure 23. SVM, CNN, and H-CNN mistake the data of fault (Figure 23(d)) as normal data probability is 25.15%, 8.42%, and 4.73%, respectively.

### Table 9. Comparison result of Huffman-convolutional neural network (H-CNN) and other methods in case 2 (%).

| Methods      | Average accuracy Known/Unknown |
|--------------|--------------------------------|
| H-CNN        | 98.62/95.27                   |
| CNN-2D       | 98.35/91.58                   |
| CNN-1D       | 90.74/82.29                   |
| DBN          | 85.48/70.43                   |
| LSTM         | 93.24/74.28                   |
| ResNet       | 91.82/78.93                   |
| RF           | 85.74/76.53                   |
| KNN          | 77.68/63.59                   |
| SVM          | 86.25/74.85                   |
| ANN          | 70.24/63.65                   |

ANN: artificial neural network; CNN: convolutional neural network; DBN: deep belief network; H-CNN: Huffman-convolutional neural network; KNN: k-nearest neighbor; RF: random forest; SVM: support vector machine.

**Figure 23.** The confusion matrices of different methods in case 2: (a) SVM, (b) CNN, and (c) H-CNN. CNN: convolutional neural network; H-CNN: Huffman-convolutional neural network; SVM: support vector machine.
Case 3: Gearbox system. The images (60 × 60) converted from gearbox system raw signals are shown in Figure 24.

As shown in Table 5, the training data contains seven classes of data and the testing data contains nine classes of data. The accuracy of the recognition is defined as follows:

\[
\text{Accuracy} = \begin{cases} 
\frac{N_{\text{predict}}(\text{Label}_\text{predict} = \text{Label}_\text{real})}{N_{\text{label}:1-7}} & \text{Label: 1 – 7} \\
\frac{N_{\text{predict}}(\text{Label}_\text{predict} \neq 1)}{N_{\text{label}:8-9}} & \text{Label: 8 – 9}
\end{cases}
\]

The results of case 3 are shown in Table 10.

Table 10. Comparison result of Huffman-convolutional neural network (H-CNN) and other methods in case 3 (%).

| Methods     | Average accuracy | Known/Unknown |
|-------------|------------------|---------------|
| H-CNN       | 99.79/96.32      |               |
| CNN-2D      | 98.63/93.74      |               |
| CNN-1D      | 93.13/80.42      |               |
| DBN         | 80.67/71.38      |               |
| LSTM        | 90.28/75.14      |               |
| ResNet      | 92.44/69.58      |               |
| RF          | 83.57/78.62      |               |
| KNN         | 75.34/62.97      |               |
| SVM         | 89.76/77.25      |               |
| ANN         | 71.38/65.82      |               |

ANN: artificial neural network; CNN: convolutional neural network; DBN: deep belief network; H-CNN: Huffman-convolutional neural network; KNN: k-nearest neighbor; RF: random forest; SVM: support vector machine.

Figure 24. Gearbox raw signals conversion results.
From Table 10, the proposed H-CNN method obtains a better result compared with other methods in this case. The average prediction accuracy is as high as 99.79% for known data and 96.32% for unknown data. For the known data, the prediction results of CNN-2D, CNN-1D, DBN, LSTM, ResNet, RF, KNN, SVM, and ANN are 98.63%, 93.13%, 80.67%, 90.28%, 92.44%, 83.57%, 75.34%, 89.76%, and 71.38%, respectively. As for unknown data, the results of CNN-2D, CNN-1D, DBN, LSTM, ResNet, RF, KNN, SVM, and ANN are 93.74%, 80.42%, 71.38%, 75.14%, 69.58%, 78.62%, 62.97%, 77.25%, and 65.82%, respectively.

The confusion matrices of different methods in case 3 are shown in Figure 25. SVM, CNN, and H-CNN mistake the data of fault (chip 2a) as normal data probability is

![Confusion Matrices](image)

**Figure 25.** The confusion matrices of different methods in case 1: (a) SVM, (b) CNN, and (c) H-CNN.

CNN: convolutional neural network; H-CNN: Huffman-convolutional neural network; SVM: support vector machine.

From Table 10, the proposed H-CNN method obtains a better result compared with other methods in this case. The average prediction accuracy is as high as 99.79% for known data and 96.32% for unknown data. For the known data, the prediction results of CNN-2D, CNN-1D, DBN, LSTM, ResNet, RF, KNN, SVM, and ANN are 98.63%, 93.13%, 80.67%, 90.28%, 92.44%, 83.57%, 75.34%, 89.76%, and 71.38%, respectively. As for unknown data, the results of CNN-2D, CNN-1D, DBN, LSTM, ResNet, RF, KNN, SVM, and ANN are 93.74%, 80.42%, 71.38%, 75.14%, 69.58%, 78.62%, 62.97%, 77.25%, and 65.82%, respectively.

The confusion matrices of different methods in case 3 are shown in Figure 25. SVM, CNN, and H-CNN mistake the data of fault (chip 2a) as normal data probability is
24.47%, 5.14%, and 3.54%, respectively. As for the fault (chip 1a), the misjudgment rates are 21.03%, 7.38% and 3.82%, respectively.

In addition, as can be seen from Figure 25, since chip 2a and chip 1a belong to the same type of faults with different degrees, the three methods mentioned above are easy to identify them as chip-*a type. Moreover, the closer the fault degree is, the less accurate these identification methods can be in distinguishing them.

**Conclusion**

In this research, we mainly discuss the generalization ability of CNN when handling fault diagnosis tasks with more test data categories than training data categories. A new fault diagnosis method called H-CNN is proposed based on CNN and Huffman pooling for rotating assembly and a new objective function based on the mixed loss is designed for the H-CNN. Then, we apply the proposed method in three different datasets. To verify the effectiveness of the method proposed in this article, comparisons are made with some other state-of-the-art methods, including CNN-2D, CNN-1D, DBN, LSTM, ResNet, RF, KNN, SVM, and ANN. The average accuracy of classifying known faults under various working conditions are H-CNN—97.93%, CNN-2D—98.26%, CNN-1D—92.14%, DBN—84.84%, LSTM—91.71%, ResNet—92.37%, RF—83.65%, KNN—75.14%, SVM—84.55%, and ANN—70.02%. The average accuracy for unknown anomaly detection under various working conditions are H-CNN—94.51%, CNN—2D—91.33%, CNN—1D—80.34%, DBN—71.68%, LSTM—71.66%, ResNet—74.25%, RF—75.54%, KNN—64.13%, SVM—74.15%, and ANN—63.03%. Compared with other methods, the results show that the proposed method has a good performance in improving the accuracy of classifying the known data and detecting the unknown fault data as anomaly data. At the same time, compared with other advanced domain generalization methods, the proposed method does not need to adopt other complex auxiliary methods to extract fault features in advance, and only needs to use the simplest CNN network structure to achieve better generalization effect. Therefore, the method proposed in this article has a good development prospect in the field of fault diagnosis of rotating component of the mechanical equipment.

**Declaration of conflicting interests**

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

**Funding**

The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Key R&D Program of China, National Natural Science Foundation of China and Science Research Project (grant numbers 2021YFA1003501, 52075117 and JSZL2020203B004).

**ORCID iD**

Mingjia Lei https://orcid.org/0000-0002-1644-8059
References

1. Wang X, Si S and Li Y. Multiscale diversity entropy: a novel dynamical measure for fault diagnosis of rotating machinery. *IEEE Trans Ind Inf* 2020; 17: 5419–5429.

2. Yang Y, Zheng H, Yin J, et al. Refined composite multivariate multiscale symbolic dynamic entropy and its application to fault diagnosis of rotating machine. *Measurement* 2019; 151: 206–216.

3. Yin J, Xu M and Zheng H. Fault diagnosis of bearing based on symbolic aggregate approximation and Lempel-Ziv. *Measurement* 2019; 151: 206–216.

4. Gao Z, Cecati C and Ding SX. A survey of fault diagnosis and fault-tolerant techniques-part I: fault diagnosis with model-based and signal-based approaches. *IEEE Trans Ind Electron* 2015; 62: 3757–3767.

5. S YL and Z S and, et al. L. Review of local mean decomposition and its application in fault diagnosis of rotating machinery[J]. *J Syst Eng Electron* 2019; 30: 799–814.

6. Wei Y, Yang Y, Xu M, et al. Intelligent fault diagnosis of planetary gearbox based on refined composite hierarchical fuzzy entropy and random forest[J]. *ISA Trans* 2020; 109: 340–351.

7. Hu Q, Qin A, Zhang Q, et al. Fault diagnosis based on weighted extreme learning machine with wavelet packet decomposition and KPCA. *IEEE Sensors J* 2018; 18: 8472–8483.

8. Zhou W, Habetler TG and Harley RG. Bearing fault detection via stator current noise cancellation and statistical control. *IEEE Trans Ind Electron* 2008; 55: 4260–4269.

9. Parey A and Pachori RB. Variable cosine windowing of intrinsic mode functions: application to gear fault diagnosis. *Measurement* 2012; 45: 415–426.

10. Chen X and Feng Z. Time-frequency analysis of torsional vibration signals in resonance region for planetary gearbox fault diagnosis under variable speed conditions. *IEEE Access* 2017; 5: 21918–21926.

11. Wang D and Peter WT. Prognostics of slurry pumps based on a moving-average wear degradation index and a general sequential Monte Carlo method. *Mech Syst Signal Process* 2015; 56: 213–229.

12. Mba C, Makis V, Marchesiello S, et al. Condition monitoring and state classification of gearboxes using stochastic resonance and hidden Markov models. *Measurement* 2018; 126: 76–95.

13. Lin Q, Zhang Y, Yang S, et al. A self-learning and self-optimizing framework for the fault diagnosis knowledge base in a workshop, Robot. *CIM-INT Manuf* 2020; 65: 101975.

14. Zhang Y, Ye D and Liu Y. Robust locally linear embedding algorithm for machinery fault diagnosis. *Neurocomputing* 2018; 273: 323–332.

15. Chen G, Liu F and Huang W. Sparse discriminant manifold projections for bearing fault diagnosis. *J Sound Vib* 2017; 399: 330–344.

16. Shao H, Jiang H, Lin Y, et al. A novel method for intelligent fault diagnosis of rolling bearings using ensemble deep auto-encoders. *Mech Syst Signal Process* 2018; 102: 278–297.

17. Su Z, Wang F, Xiao H, et al. A fault diagnosis model based on singular value manifold features, optimized SVMs and multi-sensor information fusion. *Meas Sci Technol* 2020; 31: 095002.

18. Shao H, Jiang H, Zhang X, et al. Rolling bearing fault diagnosis using an optimization deep belief network. *Meas Sci Technol* 2015; 26: 115002.

19. Ince T, Kiranyaz S, Eren L, et al. Real-time motor fault detection by 1-D convolutional neural networks. *IEEE Trans Ind Electron* 2016; 63: 7067–7075.

20. Wen L, Gao L and Li XY. A new deep transfer learning based on sparse auto-encoder for fault diagnosis. *IEEE Trans. Syst., Man, Cybern, Syst* 2017; 49: 136–144.

21. Lu C, Wang Z, Qin W, et al. Fault diagnosis of rotary machinery components using a stacked denoising autoencoder-based health state identification. *Signal Process* 2017; 130: 377–388.
22. Lei Y, Jia F, Lin J, et al. An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data. *IEEE Trans Ind Electron* 2016; 63: 3137–3147.

23. Kumar A, Gandhi C, Zhou Y, et al. Improved deep convolution neural network (CNN) for the identification of defects in the centrifugal pump using acoustic images. *Appl Acoust* 2020; 167: 107399.

24. Jia F, Lei Y, Lu N, et al. Deep normalized convolutional neural network for imbalanced fault classification of machinery and its understanding via visualization. *Mech Syst Signal Process* 2018; 110: 349–367.

25. Zheng H, Yang Y, Yin J, et al. Deep Domain Generalization Combining A Priori Diagnosis Knowledge Toward Cross-Domain Fault Diagnosis of Rolling Bearing, in *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2021, Art No. 3501311, https://doi.org/10.1109/TIM.2020.3016068.

26. Zhang W, Li X, Ma H, et al. Open set domain adaptation in machinery fault diagnostics using instance-level weighted adversarial learning. *IEEE Trans Ind Inform* 2021; 3203. https://doi.org/10.1109/TII.2021.3054651

27. Zheng H, et al. Cross-domain fault diagnosis using knowledge transfer strategy: a review. *IEEE Access* 2019; 7: 129260–129290.

28. Tian Y, Wang ZL, Zhang LP, et al. A subspace learning-based feature fusion and open-set fault diagnosis approach for machinery components. *Adv Eng Inform* 2018; 36: 194–206.

29. Lundgren A and Jung D. Data-driven fault diagnosis analysis and open-set classification of time-series data. *Control Eng Pract* 2022; 121: 105006.

30. 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: 25.

31. Guo L, Lei Y, Xing S, et al. Deep convolutional transfer learning network: a new method for intelligent fault diagnosis of machines with unlabeled data. *IEEE Trans Ind Electron* 2019; 66: 7316–7325.

32. Guo J, Wu J, Zhang S, et al. Generative transfer learning for intelligent fault diagnosis of the wind turbine gearbox. *Sensors* 2020; 20: 1361. https://doi.org/10.3390/s20051361

33. Qin AS, Mao HL and Hu Q. Cross-domain fault diagnosis of rolling bearing using similar features-based transfer approach. *Meas J Int Meas Confed* 2021; 172: 108900.

34. Li Q, Shen C, Chen L, et al. Knowledge mapping-based adversarial domain adaptation: A novel fault diagnosis method with high generalizability under variable working conditions. *Mech Syst Sig Process* 2021; 147. https://doi.org/10.1016/j.ymssp.2020.107095

35. Yin J, Lei M, Zheng H, et al. The average coding length of Huffman coding based signal processing and its application in fault severity recognition. *Appl Sci-Basel* 2019; 9: 5051.

36. Huffman DA. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*. 1952; 40: 1098–1101. https://doi.org/10.1109/JRPROC.1952.273898

37. Case Western Reserve University Bearing Data Center Website, 2009. https://csegroups.case.edu/bearingdatacenter/home.

38. Dhalmahapatra K, Shingade R, Mahajan H, et al. Decision support system for safety improvement: an approach using multiple correspondence analysis, t-SNE algorithm and K-means clustering. *Comput Ind Eng* 2019; 128: 277–289.

39. Soetens S, Sarris A and Vansteenhuys K. A discriminative feature learning approach for deep face recognition. *Proc Eur Conf Comput Vis* 2016: 499–515. https://doi.org/10.1007/978-3-319-46478-7_31

40. Cao P, Zhang S and Tang J. Preprocessing-free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning. *IEEE Access* 2018; 6: 26241–26253.

41. Zheng H, Wang R, Yang Y, et al. Intelligent fault identification based on multisource domain generalization towards actual diagnosis scenario. *IEEE Trans Ind Electron* 2020; 67: 1293–1304.
42. Zheng H, Wang R, Yin J, et al. A new intelligent fault identification method based on transfer locality preserving projection for actual diagnosis scenario of rotating machinery. *Mech Syst Signal Process* 2020; 135: 106344.

**Author biographies**

Yuqing Li received the BE degree in mechanical design manufacturing and automation from the Harbin Institute of Technology, Harbin, China, in 2002, and the ME and PhD degrees in general mechanics from the Harbin Institute of Technology, in 2004 and 2008, respectively. He is currently an Associate Professor with the Harbin Institute of Technology. His main research interests are planning and scheduling of spacecraft, and spacecraft fault detection and diagnosis.

Mingjia Lei received the BS degree in Spacecraft Environment and Life Support Engineering in 2017, from the Harbin Institute of Technology, Harbin, China, where she is currently working toward the PhD degree in mechanics with the Deep Space Exploration Research Center. Her research interests include anomaly detection of spacecraft, fault diagnosis of machinery, intelligent fault diagnosis method, and spacecraft mission planning algorithm.

Yao Cheng received the PhD degree in general and fundamental mechanics from Harbin Institute of Technology, Harbin, China, in 2016. Since 2016, he has worked for Beijing Institute of Spacecraft System Engineering. His research interests include spacecraft reliability design and analysis.

Rixin Wang received the BE and ME degrees in computer science and the PhD degree in spacecraft design from the Harbin Institute of Technology, Harbin, China, in 1985, 1991, and 2003, respectively. He is currently an Associate Professor with the Department of Engineering Mechanics, Harbin Institute of Technology. His research interests include fault detection and diagnosis for machinery and spacecraft.

Minqiang Xu received the BE degree in electronics from Peking University, Beijing, China, in 1983, the ME degree in nuclear physics from Northeast Normal University, Changchun, China, in 1989, and the PhD degree in general and fundamental mechanics from the Harbin Institute of Technology, Harbin, China, in 1999. Since 2000, he has been a Professor with the Harbin Institute of Technology. His research interests include machinery and spacecraft fault detection and diagnosis, signal processing, and space debris modeling.