*mathematics*

MDPI

*Article*

# SMoCo: A Powerful and Efficient Method Based on Self-Supervised Learning for Fault Diagnosis of Aero-Engine Bearing under Limited Data

**Zitong Yan and Hongmei Liu ***

School of Reliability and Systems Engineering, Beihang University, Beijing 100191, China
* Correspondence: liuhongmei@buaa.edu.cn

**Abstract:** Vibration signals collected in real industrial environments are usually limited and unlabeled. In this case, fault diagnosis methods based on deep learning tend to perform poorly. Previous work mainly used the unlabeled data of the same diagnostic object to improve the diagnostic accuracy, but it did not make full use of the easily available unlabeled signals from different sources. In this study, a signal momentum contrast for unsupervised representation learning (SMoCo) based on the contrastive learning algorithm—momentum contrast for unsupervised visual representation Learning (MoCo)—is proposed. It can learn how to automatically extract fault features from unlabeled data collected from different diagnostic objects and then transfer this ability to target diagnostic tasks. On the structure, SMoCo increases the stability by adding batch normalization to the multilayer perceptron (MLP) layer of MoCo and increases the flexibility by adding a predictor to the query network. Using the data augmentation method, SMoCo performs feature extraction on vibration signals from both time and frequency domains, which is called signal multimodal learning (SML). It has been proved by experiments that after pre-training with artificially injected fault bearing data, SMoCo can learn a powerful and robust feature extractor, which can greatly improve the accuracy no matter the target diagnostic data with different working conditions, different failure modes, or even different types of equipment from the pre-training dataset. When faced with the target diagnosis task, SMoCo can achieve accuracy far better than other representative methods in only a very short time, and its excellent robustness regarding the amount of data in both the unlabeled pre-training dataset and the target diagnosis dataset as well as the strong noise demonstrates its great potential and superiority in fault diagnosis.

**Keywords:** self-supervised learning; data augmentation; limited data; fault diagnosis; aero-engine; rolling bearing

**MSC:** 90B25



**Citation:** Yan, Z.; Liu, H. SMoCo: A Powerful and Efficient Method Based on Self-Supervised Learning for Fault Diagnosis of Aero-Engine Bearing under Limited Data. *Mathematics* **2022**, *10*, 2796. https://doi.org/10.3390/math10152796

Academic Editors: Yu Jiang, Haijun Peng and Hongwei Yang

Received: 15 July 2022
Accepted: 4 August 2022
Published: 6 August 2022



## 1. Introduction

As the key component of the aero-engine rotor system, rolling bearings often work in the environment of large load and high-speed rotation, which will inevitably cause huge economic losses or safety accidents [1–3]. Therefore, it is of great significance to improve the diagnostic accuracy and efficiency of rolling bearings for the healthy and stable operation of aero-engines.

With the continuous development of artificial intelligence technology, deep learning has been widely used in rolling bearing fault diagnosis to ensure the high reliability of aero-engines [4]. However, in practical industrial situations, it is very difficult to obtain a sufficient amount of labeled data, which greatly affects the performance of fault diagnosis methods based on deep learning [5].

In this case, researchers mainly use semi-supervised learning and transfer learning to solve this problem. Semi-supervised learning uses both a large amount of unlabeled data

and a small amount of labeled data for training, thereby improving the performance of the model. A three-stage semi-supervised method using data augmentation was proposed by Yu et al. [6] for bearing fault diagnosis. Zhang et al. [7] proposed a deep generative model based on a variational autoencoder (VAE) for semi-supervised learning of bearing fault diagnosis, which can effectively utilize the dataset when only a limited part of the data has labels. Transfer learning transfers the knowledge obtained from the source domain to the target domain to improve the diagnostic performance of the target domain. Wen et al. [8] adopted a three-layer sparse autoencoder to extract the features of the original data and forced the autoencoder to create a latent feature space containing the representations of the source and target domain data by adding a maximum mean difference (MMD), thereby predicting the failure of the target domain data. Wang et al. [9] proposed a deep adversarial domain adaptation network to transfer fault diagnosis knowledge, which learns domain-invariant features from raw signals using domain adversarial training based on Wasserstein distance.

Although the above methods have achieved good results, they are still only for limited application scenarios. Specifically, for semi-supervised learning, previous work mainly uses unlabeled data of the same object, which is often difficult to obtain in practical situations. For transfer learning, it requires that the distribution difference between the source and target domain data is limited, and it requires the source domain data to be labeled [10]. In addition, when faced with different diagnostic tasks, these two methods need to use all the additional data and target diagnostic data for training, which is computationally expensive and cannot be quickly and efficiently used for various diagnostic tasks.

Unlike the above algorithms, self-supervised learning provides a new solution [11]. From the perspective of data, self-supervised learning can automatically extract meaningful features from unlabeled data for fault classification, thus making full use of the easily available unlabeled data from different sources [12–15]. From the perspective of computational efficiency, self-supervised learning can be applied to various downstream diagnostic tasks with only fine-tuning after the training is completed [16]. There is no need to reuse unlabeled data for training on various downstream tasks, so that different downstream diagnostic tasks can be quickly solved.

Contrastive learning has been successfully applied to the field of computer vision as a state-of-the-art method for self-supervised learning [17–20] by reducing the distance between different augmented views of the same image (positive pairs) and increasing the distance between augmented views of different images (negative pairs) for representation learning [21]. However, there are few studies on self-supervised learning in the field of fault diagnosis. Wang et al. [16] performed self-supervised learning by having the model identify the categories that augment the signal and convert it into a classification model. The methods based on contrastive learning include: Wei et al. [22] used the data augmentation method in the image field to perform representation learning by transforming the signal through a simple reshape based on SimCLR [18]. Ding et al. [23] used momentum contrastive learning for instance-level discrimination based on MoCo [24] for representation learning. Peng et al. [25] proposed an automatic fault feature extractor based on BYOL [21] to explore some transformations of signal time-domain features.

The above methods have made attempts to apply self-supervised learning in fault diagnosis, but the problems they address are still limited to self-supervised learning using unlabeled data from the same diagnostic object and do not take full advantage of unlabeled data that are easier to obtain in other operating conditions or even other devices. In addition, their data augmentation method is still limited to morphological changes in time-domain signals and does not take advantage of the natural multi-modal characteristics of signals, such as time-domain information and frequency-domain information.

In response to the above problems, this paper proposes a new self-supervised learning method called signal momentum contrast for unsupervised representation learning (SMoCo). It improves the original MoCo in structure and designs a sufficiently difficult task by adopting the time-domain and frequency-domain cross-learning in the data aug-

mentation stage, which helps the model to learn the essential characteristics of the signal. For more details on SMoCo, please refer to Section 3.

This paper focuses on the problem of fault diagnosis of aero-engine bearing under limited data. Based on this background, a fault diagnosis method based on SMoCo is proposed. It first performs self-supervised learning on easily accessible unlabeled data to obtain a powerful and robust feature extractor. It is worth noting that the unlabeled data can be obtained from a wide range of sources, such as laboratory data of the same model under different operating conditions, or even from completely different types of products, which greatly improves the feasibility of the method. Subsequently, the feature extractor can obtain the easily classifiable features of the target diagnostic object, thus solving the difficult problem that it is difficult to diagnose aero-engine bearing faults with little data in the actual industry. Despite its good performance, SMoCo requires a relatively long training time to learn how to extract the essential features of the signal during the self-supervised learning phase. The main contributions of this paper are summarized as follows:

1. In terms of structure, based on MoCo, this paper increases the performance of the model and the stability of training by introducing a predictor to the query network and adding batch normalization (BN) [26] to the multilayer perceptron (MLP) layer.
2. In terms of data augmentation method, this paper proposes signal multimodal learning (SML), which enables the model to learn the signal representation from both the time domain and the frequency domain, thereby characterizing the signal from two dimensions.
3. The unlabeled pre-training data used by SMoCo comes from a wide range of sources and is no longer limited to the same diagnostic object, which makes it more feasible in the real task.
4. Experiments show that SMoCo can be used as a feature extractor with fixed weights to extract robust features after pre-training on artificially injected fault bearings, whether it is a bearing with different failure modes under different working conditions or a completely different type of rolling bearing. Aero-engine high-speed rolling bearings can achieve extremely high diagnostic accuracy with very few samples, providing timelier and more robust fault diagnosis than other state-of-the-art techniques.
5. Further studies have shown that SMoCo can still achieve excellent performance with a much-reduced data volume and in the presence of strong noise, further broadening its applicability.

The paper is structured as follows. In Section 2, the structure and idea of the original MoCo are introduced. In Section 3, the SMoCo algorithm proposed in this paper is introduced in detail, including the entire fault diagnosis process and its improvements in structure and data augmentation methods. In Section 4, the performance of SMoCo is verified via experiments on two datasets. In Section 5, this paper further explores the sensitivity of SMoCo to the size of the unlabeled pre-training dataset and its robustness to target diagnostic objects under different noise conditions. Section 6 summarizes the paper and looks at future work.

## 2. MoCo Network

MoCo [24] is a contrastive learning method with good training stability; the structure is shown in Figure 1. It performs representation learning in the latent space by minimizing the distance between different augmented views of the same data and rejecting augmented views of other samples.

MoCo uses two neural networks, query network $f_q$ and key network $f_k$, with the same structure for training, and its goal is to learn the convolutional layers in the query network to serve as feature extractors for downstream tasks. Since negative pair-based contrastive learning relies on the number of negative samples for representation learning, MoCo maintains a queue, which contains a single positive sample and multiple negative samples, the model learns representations by finding the corresponding positive samples.

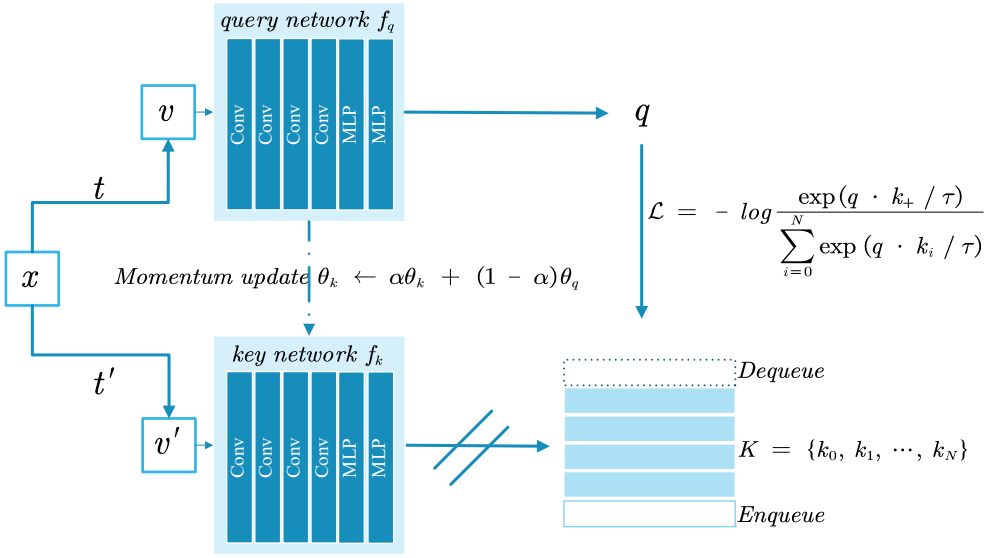



**Figure 1.** The framework of the MoCo network.

Specifically, for a given sample $x$ and the distribution $T$ of its data augmentation methods, the data augmentation methods $t \sim T$ and $t' \sim T$ are adopted respectively to generate two different augmented views of the same instance, denoted as $v$ and $v'$, and treat these two as a positive pair. Input $v$ to the $f_q$ produces a query batch $q$, and input $v'$ to the $f_k$ produces the features in the queue. It uses a dynamically updated queue to store the representations of multiple batches recently used for training. After a new batch enters the queue, the oldest training batch is out of the queue, thereby maintaining a large number of negative samples to help model training. For a given set of queues, $K = \{k_0, k_1, \cdots, k_N\}$ contains N + 1 encoding keys, where the encoder $f_k$ produces a positive sample $k_+$ for the current $v'$, and the others are negative samples, thus transforming the contrastive learning task into positive and negative samples corresponding to a given query $q$. Finally, InfoNCE [27] is used as the loss function:

$$\mathcal{L} = -\log \frac{\exp(q \cdot k_+ / \tau)}{\sum_{i=0}^{N} \exp(q \cdot k_i / \tau)} \tag{1}$$

where $\tau$ is the temperature parameter.

During the training process, only the parameters of the $f_q$ are updated via gradient back-propagation, while the parameters of the $f_k$ are updated via a momentum update. Specifically, denoting the parameters of $f_k$ as $\theta_k$ and the parameters of $f_q$ as $\theta_q$, it updates $\theta_k$ by:

$$\theta_k \leftarrow \alpha\theta_k + (1-\alpha)\theta_q \tag{2}$$

where $\alpha \in [0, 1)$ is the momentum update parameter.

MoCo builds a dynamic dictionary by using queues and momentum updates, which enables it to learn in a wider range of negative samples, making the network learn better and train more stably.

## 3. SMoCo

The framework of fault diagnosis based on SMoCo is shown in Figure 2, which is mainly composed of three key steps: (1) data acquisition, (2) self-supervised on unlabeled data, (3) fault diagnosis on labeled data. Given the difficulty of fault diagnosis of aero-engines in the case of limited data, we use unlabeled vibration signals that are easily obtained from different working conditions or even different equipment. Self-supervised learning is first employed with unlabeled signals, using our proposed signal multimodal learning (SML) as the data augmentation method. After the training is completed, the

convolutional layers of the query network are selected as the feature extractor for the downstream task, and it is worth noting that its weights remain unchanged. Finally, for the downstream labeled aero-engine bearing dataset, the feature extractor is used to extract features, and then support vector machines (SVMs) are used to classify the extracted features, and finally, the diagnostic model is obtained. The SVM is a classifier that classifies data in a supervised learning manner, where the decision boundary is the maximum-margin hyperplane solved for the learned samples.

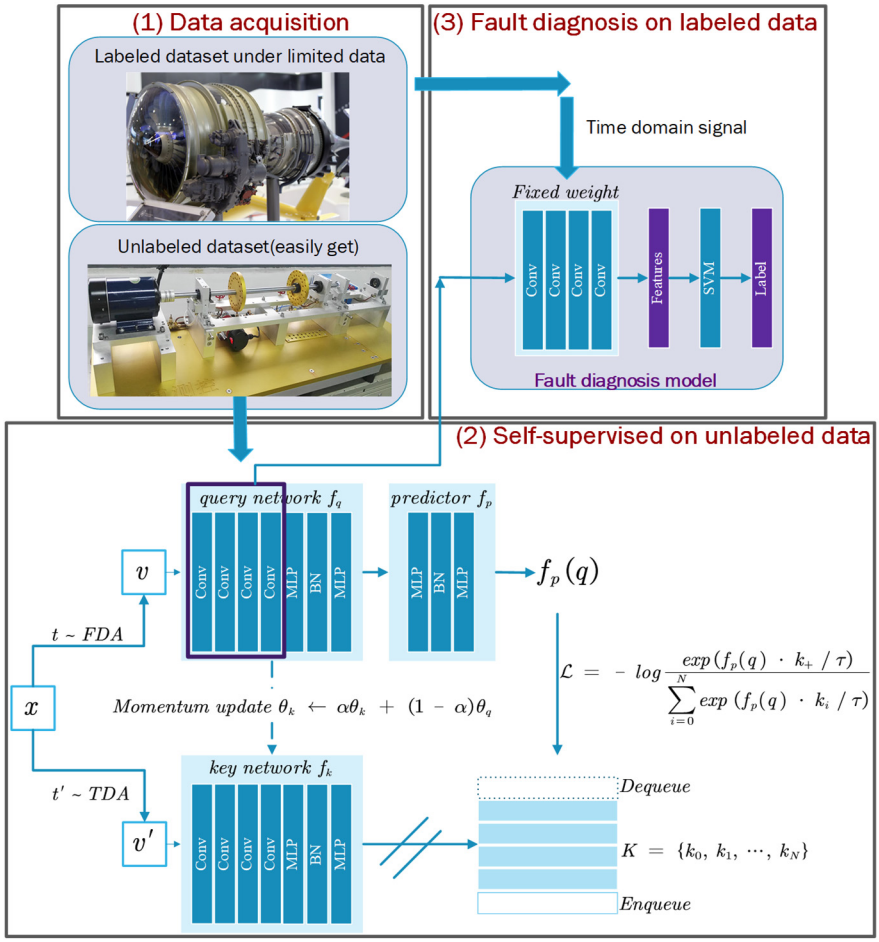



**Figure 2.** The framework of fault diagnosis based on SMoCo.

In this section, we first describe our unique data augmentation approach, signal multimodal learning (SML). Then, the network structure of SMoCo is proposed through several improvements based on MoCo. Finally, we specify an implementation detail based on SMoCo for fault diagnosis.

### *3.1. Signal Multimodal Learning (SML)*

The representation learning ability of contrastive learning depends greatly on the design and optimization of data augmentation methods [18]. Aero-engine rolling bearings diagnosis has difficult problems such as variable working conditions, strong noise, and weak faults in a real task. If a model can be unaffected by these factors, then the essential characteristics of the signal can be well characterized. The previous work was only limited to making some morphological changes to the time-domain signal when designing data augmentation methods. This paper proposes SML from the perspective of the time domain and the frequency domain according to the characteristics of vibration signals, including six basic data augmentation transformations as shown in Figure 3. The following describes in detail how these methods transform a given vibration signal $x = [x_1, x_2, \cdots, x_N]$.

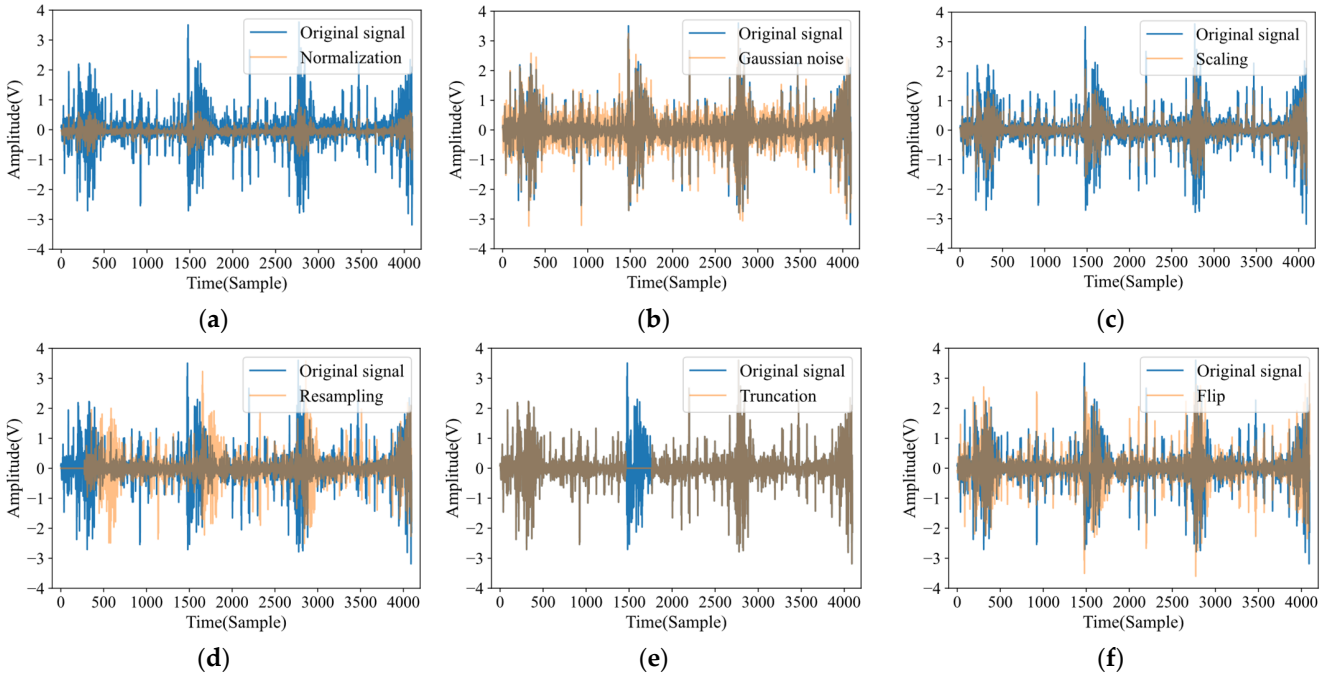



**Figure 3.** Data augmentation of sampled signals. (**a**) Normalization; (**b**) Gaussian noise; (**c**) scaling; (**d**) resampling; (**e**) truncation; (**f**) flip.

1. Normalization: There are differences in the measurement range of different sensors. This strategy normalizes the signal to a uniform range, which is also beneficial for model training. The formula is as follows:

$$\widetilde{x} = -1 + 2 * \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3}$$

2. Gaussian noise: There is an inevitable environmental noise problem during the operation of the device. This strategy adds Gaussian noise to the original signal to mimic this phenomenon. The formula is as follows:

$$\widetilde{x} = x + n, n \sim N(0, \sigma_n) \tag{4}$$

where $n$ is generated by the Gaussian distribution $N(0, \sigma_n)$.

3. Scaling: This strategy increases the sensitivity of the model to signals of different amplitudes by directly amplifying or reducing the amplitude of the signal without losing the semantics contained in the original data. The formula is as follows:

$$\widetilde{x} = x * s, s \sim N(1, \sigma_s) \tag{5}$$

where $s$ is generated from a Gaussian distribution $N(1, \sigma_s)$.

4. Resampling: This strategy improves the robustness of the model to variable speed scenarios by resampling and transforming the signal length to $s \sim N(1, \sigma_s)$ times the original length.

5. Truncation: This strategy randomly covers part of the signal, and its formula is as follows:

$$\widetilde{x} = x * mask \tag{6}$$

where *mask* is a binary sequence with subsequence zeros at random positions.

6. Flip: The vibration signal usually vibrates up and down with 0 as the mean value. This strategy randomly flips the signal to increase the diversity of the signal. The formula is as follows:

$$\widetilde{x} = -x \tag{7}$$

Since the signal naturally has multi-modal characteristics, our proposed SML treats the time-domain signal and the frequency-domain signal using fast Fourier transform (FFT) as a positive pair, as shown in Figure 4. If the model can correspond the augmented time-domain signal to the augmented frequency-domain signal, it can characterize the signal more comprehensively from both the time-domain and frequency-domain dimensions. Specifically, according to the characteristics of the time-domain signal, the order of normalization, Gaussian noise, scaling, resampling, truncation, and flip is used as the data augmentation method, which is called time-domain augmentation (TDA). For the frequency-domain signal, the order of normalization and Gaussian noise is used as the data augmentation method, which is called frequency-domain augmentation (FDA).

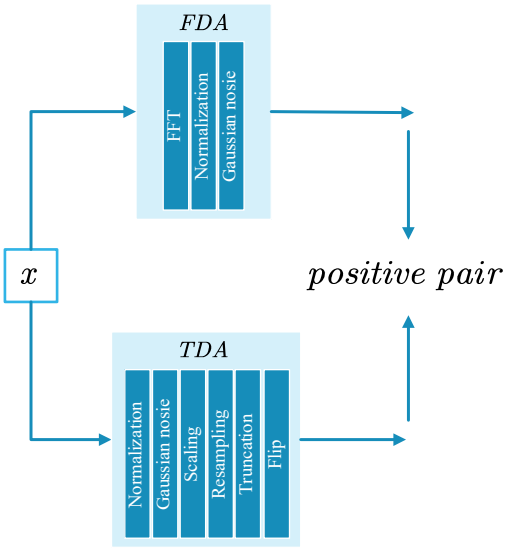



**Figure 4.** Signal multimodal learning (SML).

### *3.2. Fault Diagnosis Based on SMoCo*

The network structure of SMoCo is shown in Figure 5, which includes query network $f_q$, predictor $f_p$, key network $f_k$, and *queue*. The query network and the key network have the same structure. To increase the stability of model training, we add BN to the MLP projection layer based on MoCo. In addition, we add a predictor to the query network, which greatly increases the flexibility, so that the characteristics of the query network do not need to be the same as those of the key network, but only need to be matched by another predictor, which greatly improves the effect of representation learning. Like MoCo, SMoCo maintains a dynamically updated queue, using only the gradient to update $f_q$, and using the parameter of $f_q$ to momentum update the parameters of $f_k$. Specifically, denoting the parameters of $f_k$ as $\theta_k$ and the parameters of $f_q$ as $\theta_q$, it updates $\theta_k$ according to the Equation (2).

Given a vibration signal $x$, the data augmentation distribution of the time-domain signal is TDA, and the data augmentation distribution of the frequency-domain signal is FDA. By adopting the data augmentation strategies $t \sim FDA$ and $t' \sim TDA$ for $x$, two augmented time series $v = t(FFT(x))$ and $v' = t'(x)$ are generated. For $v$, use the query network to output the feature $q = f_q(v)$, and then use the predictor to predict $q$ to get $f_p(q)$. For $v'$, the key network outputs $k_+ = f_k(v')$. Therefore, for a given queue, for $f_p(q)$, except

for $k_+$, which is a positive pair, all other features in the queue are negative pairs. Its loss function is formulated as:

$$\mathcal{L} = -log\frac{exp(f_p(q)\cdot k_+/\tau)}{\sum_{i=0}^{N} exp(f_p(q)\cdot k_i/\tau)} \tag{8}$$

where $\tau$ is the temperature parameter.

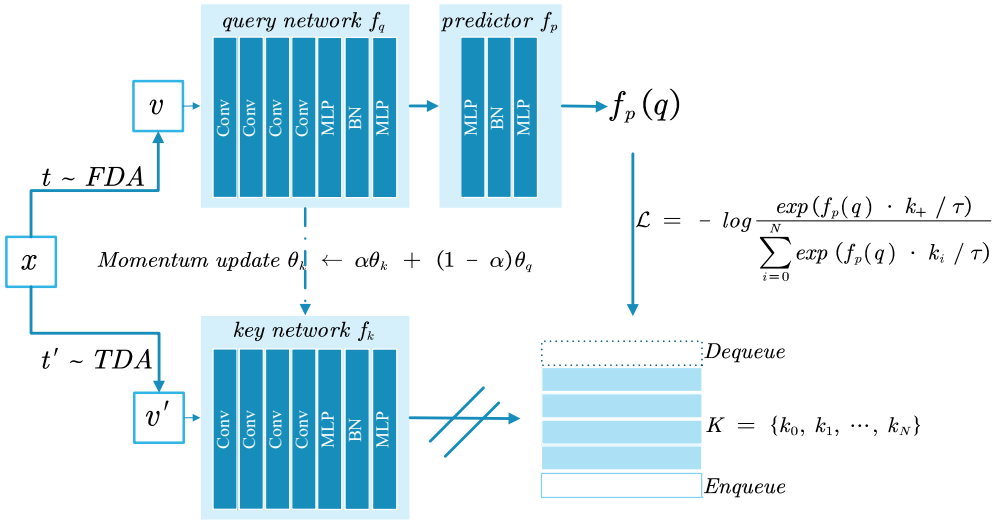



**Figure 5.** The structure of SMoCo.

It gets the symmetric loss function $\widetilde{\mathcal{L}}$ by feeding $v'$ to the query network and $v$ to the key network. Finally, the network updates the query network $f_q$ by minimizing the loss $\mathcal{L}_{SMoCo}$:

$$\mathcal{L}_{SMoCo} = 0.5 \times \left(\mathcal{L} + \widetilde{\mathcal{L}}\right) \tag{9}$$

The detailed SMoCo is shown in Algorithm 1.

**Algorithm 1.** The detailed SMoCo.

**Input:**
Structure of $f_q$, $f_p$, $f_k$, temperature $\tau$, momentum update $\alpha$, *queue* size $N$
batch size $n_b$, learning rate $\eta$, total number of optimization steps $K$,
distributions of transformations TDA, FDA, set of signals D
Initialize parameters, $\theta_k \leftarrow \theta_q$, and *queue*
**for** $k = 1$ to $K$ **do**
  Batch $\leftarrow \{x_i \sim D\}_{i=1}^{n_b}$
  **for** $x_i \in$ Batch **do**
    $t \in FDA$ and $t' \in TDA$
    $q^1 \leftarrow f_q(t(x_i))$ and $k_+^1 \leftarrow f_k(t'(x_i))$
    $q^2 \leftarrow f_q(t'(x_i))$ and $k_+^2 \leftarrow f_k(t(x_i))$
    $l_i \leftarrow 0.5 \times (-\log\frac{\exp(f_p(q^1)\cdot k_+^1/\tau)}{\sum_{i=0}^{N}\exp(f_p(q^1)\cdot k_i/\tau)} - \log\frac{\exp(f_p(q^2)\cdot k_+^2/\tau)}{\sum_{i=0}^{N}\exp(f_p(q^2)\cdot k_i/\tau)})$
  **end**
    // Back-propagation
  $\theta_q \leftarrow \theta_q - \eta\cdot\frac{\partial\frac{1}{n_b}\sum_{i=1}^{n_b} l_i}{\theta_q}$
    // Momentum update without back-propagation
  $\theta_k \leftarrow \alpha\theta_k + (1-\alpha)\theta_q$
    // Update dictionary
    Enqueue and dequeue with $\{k_+^1\}_{i=1}^{n_b}$ and $\{k_+^2\}_{i=1}^{n_b}$
**end**
**Output:** query network parameters $\theta_q$

After training, the convolutional layers in the query network are extracted to perform feature extraction on downstream tasks. When performing downstream tasks, the weights of the convolutional layers remain fixed and only serve as a function of feature extraction. Since the SVM is the classifier with the largest interval in the feature space, the SVM is adopted to classify the extracted features, which is more robust in the problem under limited data.

## 4. Performance Verification of SMoCo

To verify the effectiveness and superiority of SMoCo, as proposed in this paper, the bearing dataset of Paderborn University and the aero-engine bearing dataset of the Polytechnic University of Turin are used for experimental verification. SMoCo is first pre-trained on the unlabeled laboratory data of artificially injected faults from Paderborn University. The learned feature extractors are then transferred to products of the same type but with failures generated in natural operation from Paderborn University, and these two datasets are characterized using different working conditions, different failure levels, and different failure modes. It is further transferred to the aero-engine bearing dataset from the Polytechnic University of Turin, which is a completely different model compared to the pre-training dataset, and the data distributions of these two datasets differ significantly and thus can effectively verify the validity of aero-engine bearing fault diagnosis under limited data. The purpose of using two cases is to verify the effect of the proposed method on different diagnostic subjects.

### *4.1. Self-Supervised on Artificially Damaged Bearing Data*

The Paderborn University dataset [28] is a public dataset collected by the Paderborn University Bearing Data Center in 2016 with high diagnostic difficulty [29]. In this dataset, bearing damages are rich and can be divided into three categories: 6 healthy bearings, 12 artificially damaged bearings, and 14 real damaged bearings. Among them, the real damaged data were obtained through the accelerated life test. The vibration signal was obtained at a sampling rate of 64 khz, including 4 working conditions, as shown in Table 1, and the test rig is shown in Figure 6.

**Table 1.** Operating parameters.

| Name of Setting | Rotational Speed [rpm] | Load Torque [Nm] | Radial Force [N] |
|---|---|---|---|
| N09_M07_F10 | 900 | 0.7 | 1000 |
| N15_M07_F10 | 1500 | 0.7 | 1000 |
| N15_M01_F10 | 1500 | 0.1 | 1000 |
| N15_M07_F04 | 1500 | 0.7 | 400 |

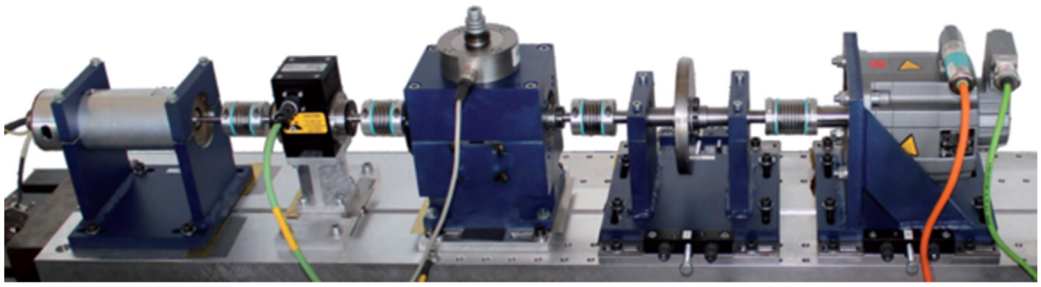


**Figure 6.** Test rig of Paderborn dataset.

To better represent the easy-to-obtain unlabeled data, the artificially injected fault bearing data in the Paderborn University dataset is used as the unlabeled pre-training dataset, as shown in Table 2. There are 13 types of bearings including one type of health status; 4096 is selected as the sample length to contain enough information, and the working condition is N15_M01_F10. The number of samples in each category is 2000, and all data are kept as raw time-domain data without any signal pre-processing.

**Table 2.** Dataset 1: Unlabeled artificially damaged bearing dataset under N15_M01_F10.

| Bearing Code | Damaged Element | Damaged Extent | Damage Method |
|---|---|---|---|
| K001 | Health state | / | Run-in 50 h before test |
| KA01 | Outer ring | Level 1 | Made by EDM |
| KA03 | Outer ring | Level 2 | Made by electric engraver |
| KA05 | Outer ring | Level 1 | Made by electric engraver |
| KA06 | Outer ring | Level 2 | Made by electric engraver |
| KA07 | Outer ring | Level 1 | Made by drilling |
| KA08 | Outer ring | Level 2 | Made by drilling |
| KA09 | Outer ring | Level 2 | Made by drilling |
| KI01 | Inner ring | Level 1 | Made by EDM |
| KI03 | Inner ring | Level 1 | Made by electric engraver |
| KI05 | Inner ring | Level 1 | Made by electric engraver |
| KI07 | Inner ring | Level 2 | Made by electric engraver |
| KI08 | Inner ring | Level 2 | Made by electric engraver |

The original MoCo used ResNet50 [30] as the backbone network and achieved excellent results. However, as the number of network layers increases, the computational complexity of the network gradually increases and it is difficult to converge. The original MoCo uses a deep ResNet network because it is used to solve computer vision tasks, while the feature learning task of bearings is less difficult than the feature learning task of images, so the backbone network of SMoCo adopts the ResNet18 [30].

The output dimension of the query network and the key network is 128 in line with MoCo, thus ensuring that there is enough space to represent the extracted features. Since the convolutional layer output of ResNet18 has a dimension of 512, the MLP layers in the query network, key network, and predictor have the same structure with a hidden layer dimension of 512 and an output layer dimension of 128, and this structure has also been shown to be very effective for representation learning [21,24].

The initial learning rate $\eta$ is set to 0.1 because using a larger learning rate can [24,25] accelerate the convergence and allows the model to try multiple directions at the early stage of optimization to prevent the model from getting stuck at the saddle point or the local minimum due to the small learning rate. In addition, since this paper uses both time-domain and frequency-domain data for learning, the data distribution between the two differs greatly and the learning task is more complex, therefore, 0.1 is chosen as the initial learning rate. The learning rate is updated via the cosine learning rate scheduler with the following equation.

$$\eta_t = \frac{1}{2}\left(1 + cos\left(\frac{t\pi}{T}\right)\right)\eta \tag{10}$$

where $\eta$ is the initial learning rate, $\eta_t$ is the current learning rate, $T$ is the maximum number of epochs, and $t$ is the current epoch.

It has been shown in MoCo that the model performs better when the value of momentum $\alpha$ is in the range 0.99~0.9999, showing that a slowly progressing (i.e., relatively large momentum) key encoder is beneficial, while when $\alpha$ is too small (e.g., 0.9), the accuracy drops considerably [24]. This is because MoCo relies on a consistent dictionary for training, which is the data in the queue generated by the key encoder [24]. Therefore, SMoCo chooses to keep the same parameter selection as MoCo, i.e., 0.999. See Table 3 for other hyperparameters. In addition, the data augmentation methods in Table 3 are all implemented with a probability of 50%, thereby increasing the variety of the transformation. The variation of the loss values during the training process is shown in Figure 7; it can be found that the loss value becomes smooth in the late training period, indicating that the training has reached the fitting state. The experiment was conducted under Windows 11 and PyTorch1.11, running on a computer with the following configurations: i5-12400F, NVIDIA RTX 3060, and 16GB RAM. The training time for self-supervised learning is about 6.5 h.

**Table 3.** Hyperparameter setting.

| Hyperparameter | Value | Data Augmentation | Value |
|---|---|---|---|
| Batch size | 64 | Normalization | / |
| Optimizer | SGD | Gaussian noise | Noise coefficient $\sigma_n = 0.05$ |
| Learning rate | 0.1 | Scaling | Scale coefficient $\sigma_s = 0.05$ |
| Momentum | 0.9 | Resampling | Stretch coefficient $\sigma_s = 0.3$ |
| Weight decay | 0.0001 | Truncation | Truncation length = 100 |
| Epochs | 350 | Flip | / |
| Learning rate schedule | Cosine | | |
| Queue size | 65536 | | |
| Momentum update | 0.999 | | |
| Temperature | 0.07 | | |

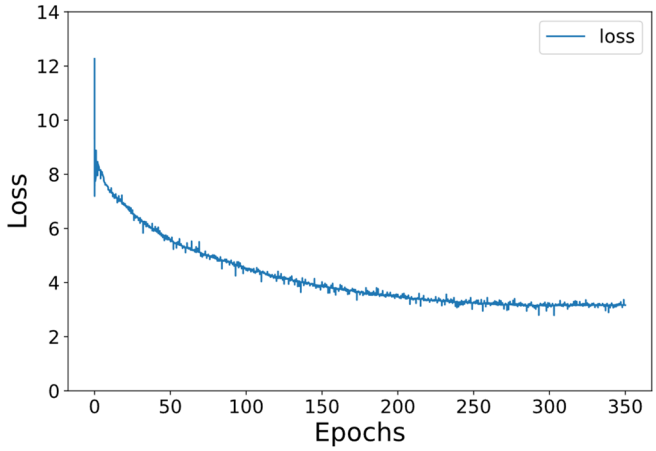



**Figure 7.** Loss history of self-supervised on unlabeled dataset 1.

Other self-supervised learning methods, Wang, SimCLR, BYOL, and MoCo, are carried out for comparison. To exclude the influence of other factors, the backbone network of all methods is ResNet18, which is trained using time-domain signals. In addition, to prove the great superiority of SMoCo, as a comparison, the labeled dataset 1 is used for supervised learning, and the network structure is also ResNet18, which is called labeled pretraining. The feature extractors of all methods, that is, the convolutional layers of ResNet18, are used to perform feature extraction on part of the data in dataset 1 and T-SNE is used to reduce it to 2D for visualization. The results are shown in Figure 8. SMoCo can achieve an excellent feature extraction performance without using labels and achieves the aggregation of each category and the separation of different categories from each other, which greatly exceeds all other self-supervised learning methods, even reaching the level of labeled pretraining. Other self-supervised methods perform poorly, specifically for Wang, which only identifies the corresponding data augmentation categories without instance-level self-supervised learning and therefore does not perform feature extraction well. For SimCLR, its reliance on learning in large batches, via comparing data within a batch without other techniques such as momentum updates, makes its training less stable and less performant. For BYOL and MoCo, although they achieve relatively good results without labels, they lack the unique SML proposed in this paper, so the results are not as good as SMoCo.

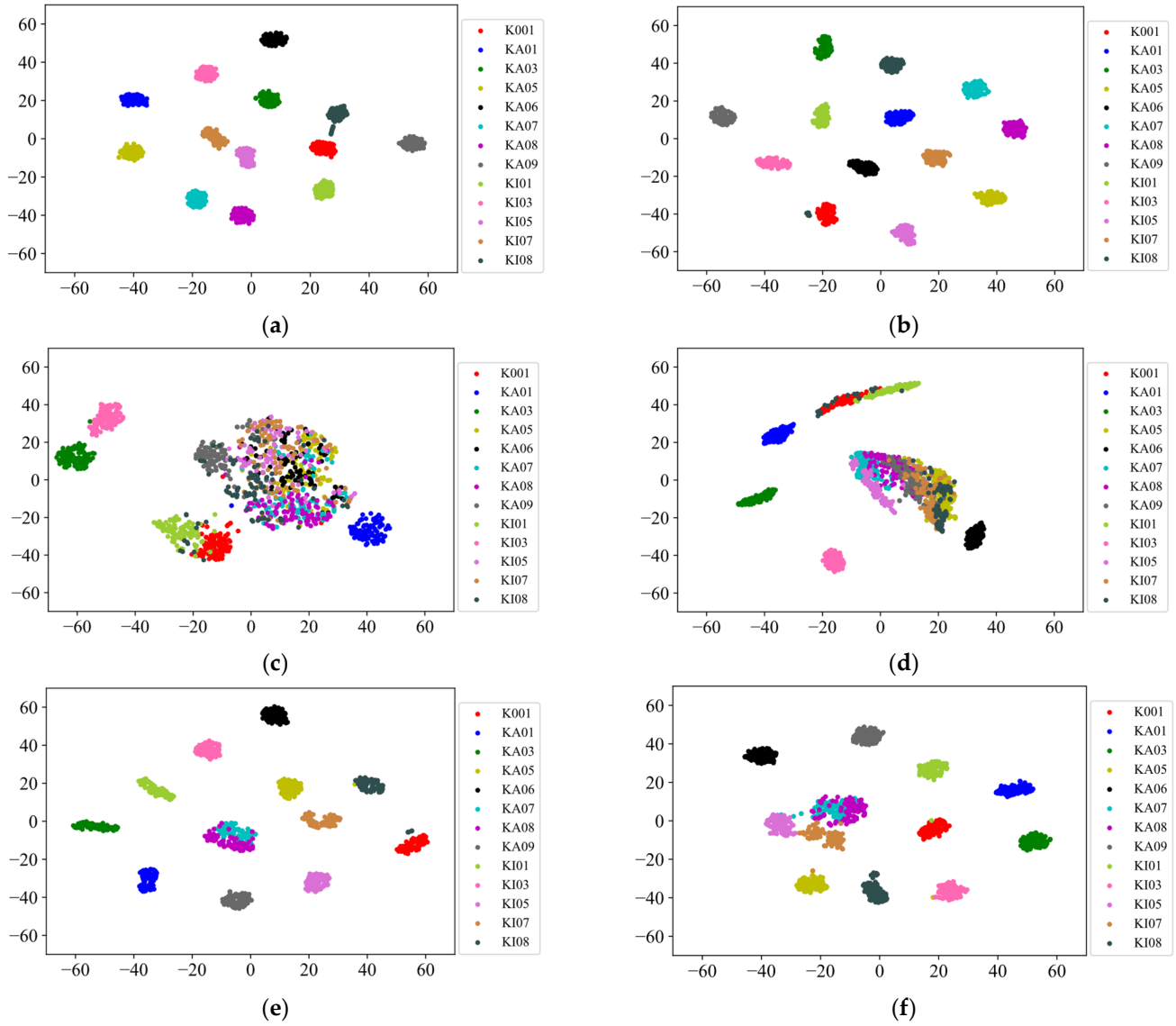



**Figure 8.** The visualization of feature extractors on unlabeled bearing dataset 1. (**a**) SMoCo; (**b**) labeled pretraining; (**c**) Wang; (**d**) SimCLR; (**e**) BYOL; (**f**) MoCo.

### *4.2. Fault Diagnosis on Same Products under Different Fault Characteristic Distributions*

To verify the diagnostic performance of SMoCo for the same products under different failure levels, different failure models, and different working conditions, 10 types of real damaged bearings in the Paderborn University dataset, including a healthy state bearing and 2 mixed fault bearings with the working condition of N15_M07_F04, are selected as the target diagnosis dataset. The specific information is shown in Table 4. To reflect the limited data problem faced in the actual diagnosis task, the training set uses 5 samples per class, and the testing set uses 50 samples per class.

To demonstrate the performance of the feature extractor obtained in the self-supervised learning stage, the feature extractors trained in Section 4.1 are used to perform feature extraction on the testing set without any training, and T-SNE is used for visualization. The results are shown in Figure 9. The SMoCo proposed in this paper can achieve an excellent feature extraction performance on target diagnostic data without using training data. It not only greatly outperforms other self-supervised learning methods, but also outperforms labeled pretraining. Compared to the result of extracting from dataset 1, other methods are less capable of extracting features from the target diagnostic data at this time

due to the difference between the distribution of the pre-training dataset and the target diagnostic dataset.

**Table 4.** Dataset 2: Real damaged bearing dataset under N15_M07_F04.

| Bearing Code | Damaged Element | Fault Mode | Damage Form | Arrangement | Damaged Extent |
|---|---|---|---|---|---|
| K001 | Health state | / | / | / | / |
| KA04 | Outer ring | Fatigue: pitting | Single damage | No repetition | Level 1 |
| KA15 | Outer ring | Plastic deform: Indentations | Single damage | No repetition | Level 1 |
| KA16 | Outer ring | Fatigue: pitting | Repetitive damage | Random | Level 2 |
| KB23 | Outer ring and inner ring | Fatigue: pitting | Multiple damage | Random | Level 2 |
| KB24 | Outer ring and inner ring | Fatigue: pitting | Multiple damage | No repetition | Level 3 |
| KI14 | Outer ring | Fatigue: pitting | Multiple damage | No repetition | Level 1 |
| KI16 | Outer ring | Fatigue: pitting | Single damage | No repetition | Level 3 |
| KI17 | Inner ring | Fatigue: pitting | Repetitive damage | Random | Level 1 |
| KI18 | Inner ring | Fatigue: pitting | Single damage | No repetition | Level 2 |

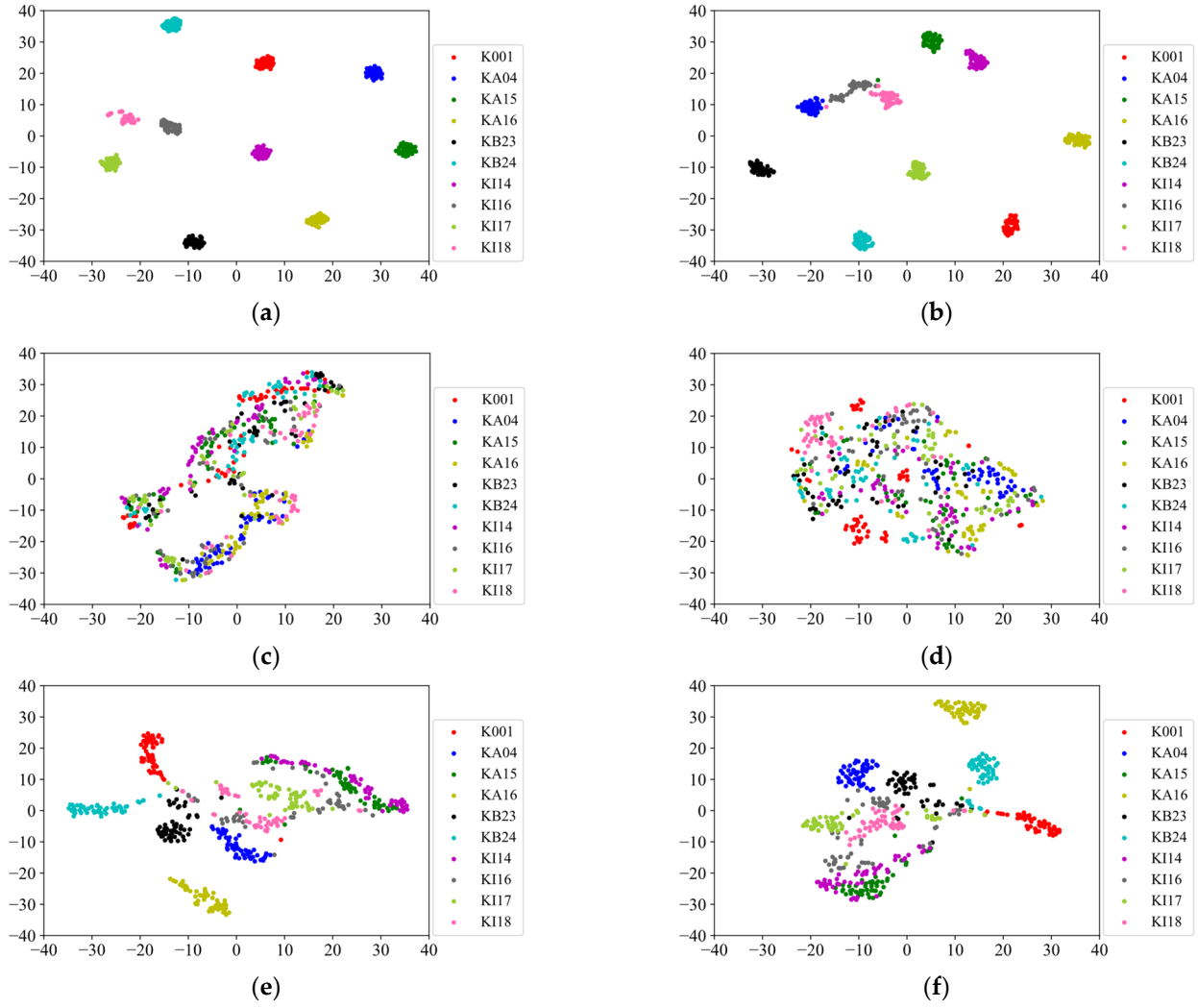



**Figure 9.** The visualization of feature extractors on labeled bearing dataset 2. (**a**) SMoCo; (**b**) labeled pretraining; (**c**) Wang; (**d**) SimCLR; (**e**) BYOL; (**f**) MoCo.

To more fully demonstrate the superiority of our method, in addition to the methods in Section 4.1, we also use MixMatch [31], ResNet18, and FFT + SVM as a comparison for

the diagnosis task. Among them, MixMatch is one of the best-performing semi-supervised methods, which uses the unlabeled dataset 1 and the training set of dataset 2 for training. ResNet18 is trained using only the training set of dataset 2. The diagnostic accuracy of each method is shown in Table 5 and Figure 10. FFT + SVM is a classical and effective fault diagnosis method for small sample cases, which first performs FFT transformation on the original signal and then uses SVM to classify the FFT transformed features.

**Table 5.** Comparison of diagnostic results on dataset 2 under 5 samples per class.

| Method | Accuracy (%) | Time (s) |
|---|---|---|
| SMoCo | 99.68 ± 0.26 | 1.47 |
| MixMatch | 89.96 ± 4.84 | 411.24 |
| Labeled Pretraining | 97.16 ± 1.80 | 25.11 |
| Wang | 73.88 ± 3.40 | 29.93 |
| SimCLR | 73.76 ± 1.35 | 30.44 |
| BYOL | 89.48 ± 3.29 | 30.72 |
| MoCo | 89.68 ± 2.16 | 30.75 |
| ResNet18 | 71.96 ± 3.13 | 26.26 |
| FFT + SVM | 79.14 ± 7.86 | 0.16 |

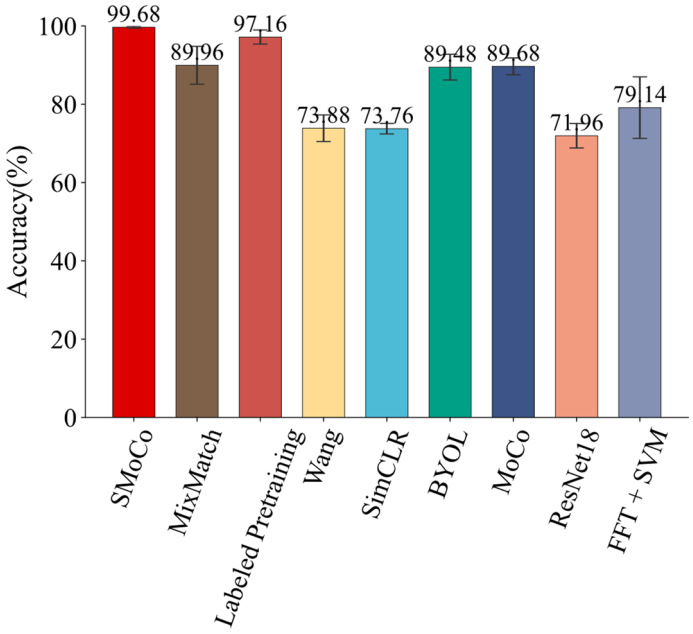
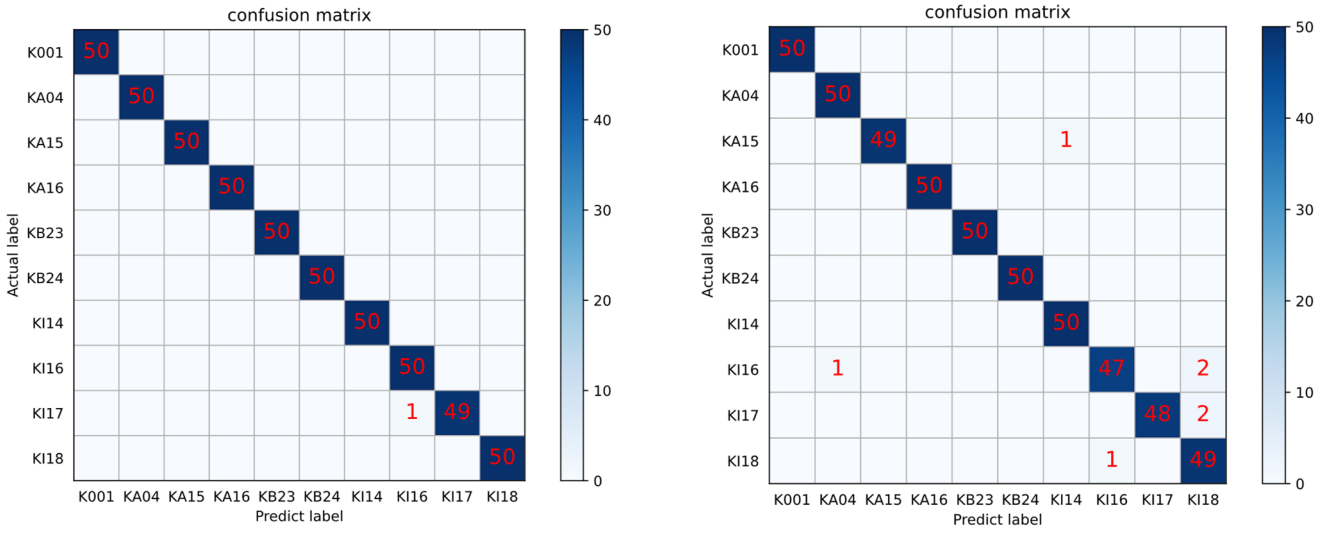



**Figure 10.** Comparison of diagnostic results on dataset 2 under 5 samples per class.

It can be seen from Table 5 and Figure 10 that SMoCo benefits from its unique SML and structural improvements to MoCo; its accuracy reaches an astonishing 99.68%, while the time it takes is only 1.47 s, which even significantly exceeds the results of labeled pretraining. This is also consistent with the visualization results in Figure 9. SMoCo can distinguish each class well before training, so it only needs to use very few samples to build an excellent classification surface. Labeled pretraining uses labels for pre-training, but the obtained feature extractor is only adapted to the pre-training dataset. When faced with new diagnostic tasks, although its diagnostic accuracy is improved, the effect is still limited. Other self-supervised learning methods lack our unique SML and gaps in the structure, so their performance falls far short of SMoCo. For FFT + SVM, it performs better than ResNet18 using only time-domain features in the case of small samples; however, its diagnostic accuracy is not high in the face of complex diagnostic problems under real faults.

The confusion matrix for SMoCo and labeled pretraining with the best diagnostic performance is plotted as shown in Figure 11. SMoCo only misclassified one sample of KI17 as KI16, which is consistent with the results visualized in Figure 9. The interval between

KI17 and KI16 is relatively close, which may cause errors in the classification plane due to the special training samples. Nonetheless, our SMoCo outperforms labeled pretraining in every category.




(**a**) (**b**)

**Figure 11.** Confusion matrix of the two best-performing methods on dataset 2. (**a**) SMoCo; (**b**) labeled pretraining.

The diagnostic accuracy of SMoCo in the case of fewer samples is also further explored by selecting the best performing SMoCo and labeled pretraining as a comparison. For the training set, a total of 5 groups of samples from 1 to 5 per class were used to explore the results, as shown in Figure 12. It can be seen from Figure 12 that SMoCo is far better than labeled pretraining in all cases. SMoCo can achieve 99.16% accuracy with only 3 samples per class and its accuracy only drops more in the case of one sample per class. It is demonstrated that our method has strong performance and robustness for diagnosis in limited data.

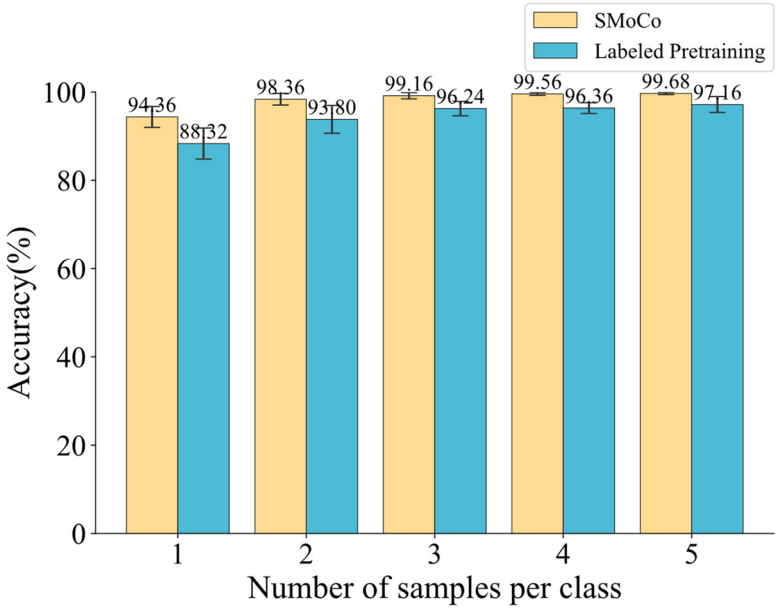



**Figure 12.** Comparison of results under different training set sizes on dataset 2.

### *4.3. Fault Diagnosis on Different Products of Aero-Engine Bearing*

To verify the diagnostic effectiveness of SMoCo on aero-engine rolling bearings, this paper uses the dataset of aero-engine high-speed bearings from the Department of Mechanical and Aeronautical Engineering of the Polytechnic University of Turin [32]. The test rig is shown in Figure 13. For the dataset, we use the vibration acceleration data of aero-engine bearings at different rotational speeds and different degrees of damage. The length of a single sample is still 4096, and the y-direction channel data at A1 is used. To reflect the extremely limited data situation in the actual diagnosis process, only 3 samples per class are used in the training set, and 50 samples per class are used in the testing set. The specific dataset information is shown in Table 6. At this point, the unlabeled pre-training dataset 1 and the target diagnostic dataset 3 have completely different device types, working conditions, and failure modes.

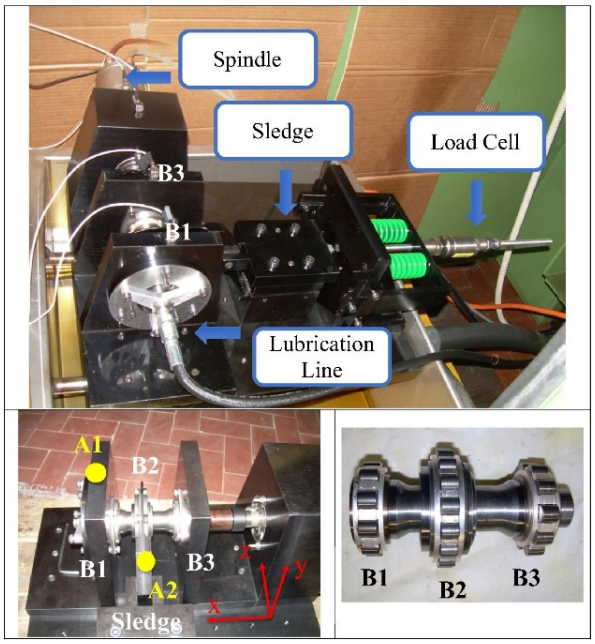



**Figure 13.** Test rig of the aero-engine bearing dataset from Polytechnic University of Turin.

**Table 6.** Dataset 3: Aero-engine bearing dataset from Polytechnic University of Turin.

| Damaged Element | Diameter (μm) | Fault Mode Rotation Speed (r/min) | Load (N) | Training Samples | Testing Samples | Label |
|---|---|---|---|---|---|---|
| Healthy | / | 24,000 | 1400 | 3 | 50 | 0 |
| Inner ring | 450 | 24,000 | 1400 | 3 | 50 | 1 |
| Inner ring | 250 | 24,000 | 1400 | 3 | 50 | 2 |
| Inner ring | 150 | 24,000 | 1400 | 3 | 50 | 3 |
| Roller | 450 | 24,000 | 1400 | 3 | 50 | 4 |
| Roller | 250 | 24,000 | 1400 | 3 | 50 | 5 |
| Roller | 150 | 24,000 | 1400 | 3 | 50 | 6 |
| Inner ring | 450 | 18,000 | 1400 | 3 | 50 | 7 |
| Inner ring | 250 | 18,000 | 1400 | 3 | 50 | 8 |
| Inner ring | 150 | 18,000 | 1400 | 3 | 50 | 9 |
| Roller | 450 | 18,000 | 1400 | 3 | 50 | 10 |
| Roller | 250 | 18,000 | 1400 | 3 | 50 | 11 |
| Roller | 150 | 18,000 | 1400 | 3 | 50 | 12 |

As in Section 4.2, the feature extractors trained in Section 4.1 are used to perform feature extraction on the testing set data and visualize it using T-SNE. It's worth noting that this was done without any training on dataset 3. The results are shown in Figure 14. The SMoCo proposed in this paper still achieves amazing feature extraction results in the face of completely different devices without using any training data, greatly surpassing other methods.

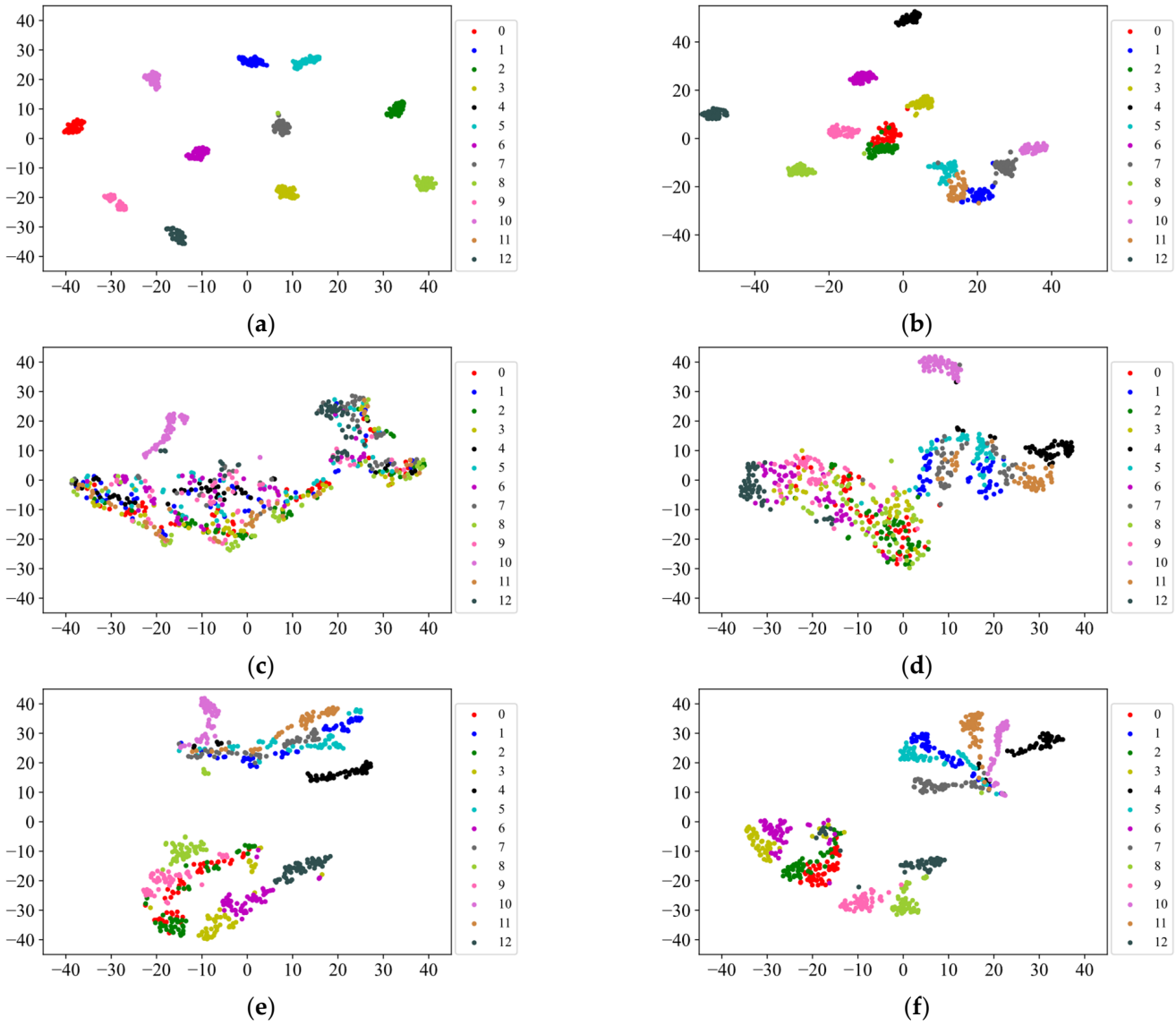



**Figure 14.** The visualization of feature extractors on labeled aero-engine bearing dataset 3. (**a**) SMoCo; (**b**) labeled pretraining; (**c**) Wang; (**d**) SimCLR; (**e**) BYOL; (**f**) MoCo.

The methods trained in Section 4.1, MixMatch, ResNet18, and FFT + SVM are used for comparison on dataset 3, and the results are shown in Table 7 and Figure 15.

It can be seen from Table 7 and Figure 15 that SMoCo achieves 100% diagnostic accuracy when faced with diagnostic problems of different devices, and its training and inference time is only 1.6 s. Both the accuracy and efficiency achieved the best results, greatly surpassing other methods. Although labeled pretraining can still improve the accuracy at this time, in the case of different devices, due to the large difference between the distribution of the pre-training data and the data to be diagnosed, namely dataset 1 and dataset 3, its effect is greatly reduced at this time. Since MixMatch uses both dataset 1 and dataset 3 for training, it can adapt the target diagnostic data with unlabeled data and thus obtain better diagnostic accuracy, but even so it is not as good as SMoCo. In addition, it

needs to be trained from scratch for each diagnostic task, so its training time is far inferior to SMoCo. The performance of other self-supervised methods is still far from that of ours. FFT + SVM has achieved good results in the face of relatively simple diagnostic tasks, but there is still a big gap compared with SMoCo.

**Table 7.** Comparison of diagnostic results on dataset 3 under 3 samples per class.

| Method | Accuracy (%) | Time (s) |
|---|---|---|
| SMoCo | 100.00 ± 0.00 | 1.60 |
| MixMatch | 98.55 ± 0.65 | 469.06 |
| Labeled Pretraining | 90.92 ± 2.11 | 30.62 |
| Wang | 74.65 ± 4.56 | 36.04 |
| SimCLR | 81.85 ± 4.06 | 37.32 |
| BYOL | 85.66 ± 2.82 | 35.12 |
| MoCo | 84.00 ± 4.10 | 40.42 |
| ResNet18 | 82.83 ± 2.88 | 29.78 |
| FFT + SVM | 94.94 ± 4.19 | 0.15 |

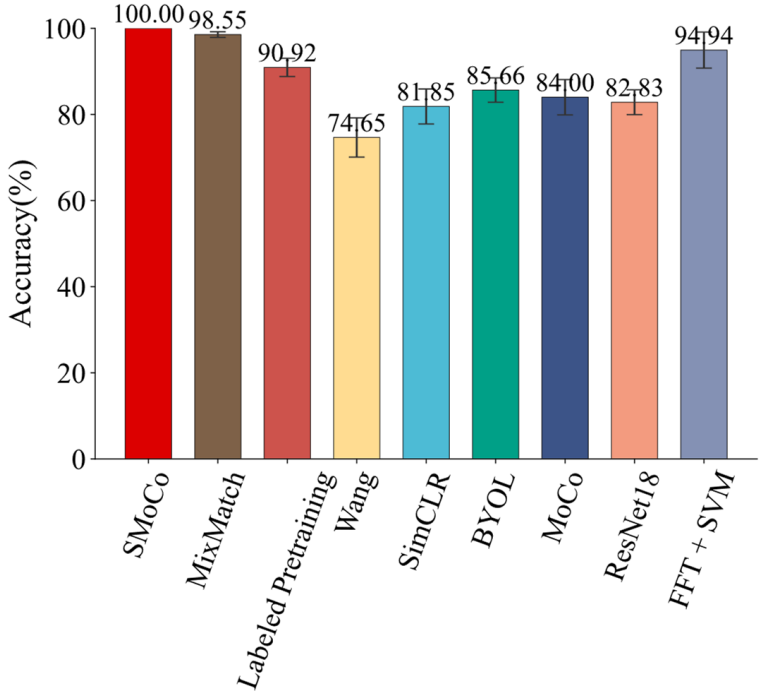



**Figure 15.** Comparison of diagnostic results on dataset 3 under 3 samples per class.

To further explore the effectiveness of SMoCo, MixMatch is also used as a comparison, which is the best performing method among the other methods. For the training set of dataset 3, one sample per class to three samples per class are used for training, and the results are shown in Figure 16. SMoCo can achieve a diagnostic accuracy of 99.15% with only one sample per class, which is also consistent with the results of feature visualization. It is proven that SMoCo is efficient and robust in the face of diagnostic tasks of different devices, which greatly reflects its superiority. In the case of extremely limited data, MixMatch's diagnostic accuracy drops sharply. This is due to the lack of a stable and efficient feature extractor, and it will encounter the common problem of deep learning, that is, the performance will be greatly reduced when the amount of data is extremely limited.

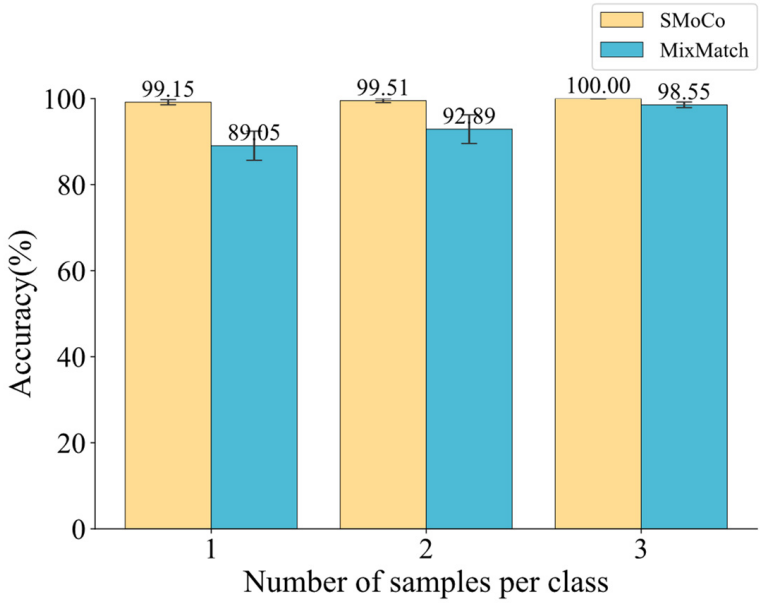



**Figure 16.** Comparison of results under different training set sizes on dataset 3.

## 5. Robustness Verification of SMoCo

### *5.1. Sensitivity to the Size of the Pre-Training Dataset*

To further explore the sensitivity of SMoCo to the size of the unlabeled pre-training dataset, in this section, five different data volumes of 2000, 1500, 1000, 500, and 100 for each class are used for self-supervised learning on unlabeled dataset 1. After the self-supervised training is completed, all feature extractors are used to perform fault diagnosis on the labeled dataset 2 and dataset 3, respectively. In addition, to further explore their performance with different numbers of labeled training sets, this paper varies the number of samples per class from 1 to 5 for the training set of dataset 2 and from 1 to 3 for dataset 3. For each dataset, an additional method that performs the best except SMoCo in Sections 4.2 and 4.3 is performed as a comparison. Finally, to better evaluate their diagnostic performance, the F1 score is used as the evaluation criterion [33], and the results are shown in Table 8, Figure 17, Table 9, and Figure 18. Where SMoCo + 2000 means self-supervised learning using 2000 unlabeled samples per class in dataset 1, the meaning of SMoCo + 1500, etc. can be deduced accordingly. Labeled pretraining + 2000 means pre-training with labels using 2000 samples per class in dataset 1. MixMatch + 2000 means semi-supervised learning using both the 2000 unlabeled samples per class in dataset 1 and the labeled target diagnostic dataset.

**Table 8.** Experimental results of the sensitivity to the size of the data set on dataset 2.

| Method | Number of Samples Per Class on Dataset 2 | | | | |
|---|---|---|---|---|---|
| | 1 (F1/%) | 2 (F1/%) | 3 (F1/%) | 4 (F1/%) | 5 (F1/%) |
| SMoCo + 2000 | 94.39 | 97.86 | 98.92 | 99.60 | 99.64 |
| SMoCo + 1500 | 92.35 | 95.43 | 97.79 | 98.52 | 98.76 |
| SMoCo + 1000 | 91.80 | 94.90 | 96.76 | 98.07 | 98.51 |
| SMoCo + 500 | 89.85 | 94.26 | 96.46 | 97.00 | 97.95 |
| SMoCo + 100 | 89.33 | 94.19 | 96.26 | 96.65 | 97.61 |
| Labeled Pretraining + 2000 | 88.33 | 94.24 | 96.20 | 96.84 | 97.20 |

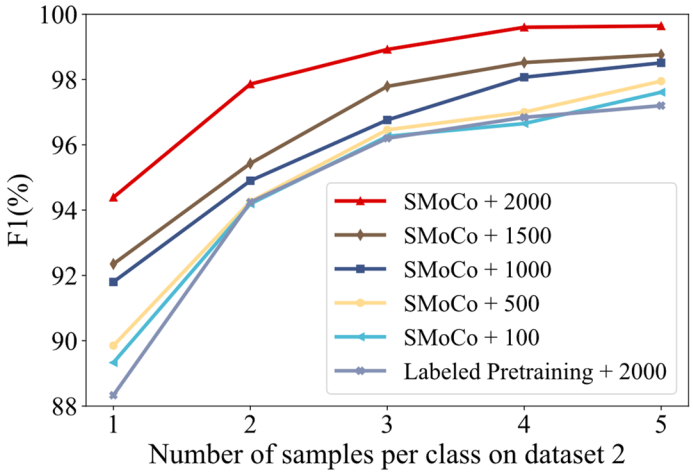



**Figure 17.** Experimental results of the sensitivity to the size of the data set on dataset 2.

**Table 9.** Experimental results of the sensitivity to the size of the data set on dataset 3.

| Method | Number of Samples Per Class on Dataset 3 | | |
|---|---|---|---|
| | 1 (F1/%) | 2 (F1/%) | 3 (F1/%) |
| SMoCo + 2000 | 99.46 | 99.84 | 99.94 |
| SMoCo + 1500 | 98.31 | 99.11 | 99.54 |
| SMoCo + 1000 | 98.01 | 99.04 | 99.38 |
| SMoCo + 500 | 97.74 | 98.86 | 99.20 |
| SMoCo + 100 | 97.04 | 98.21 | 98.89 |
| MixMatch + 2000 | 88.51 | 94.27 | 97.43 |

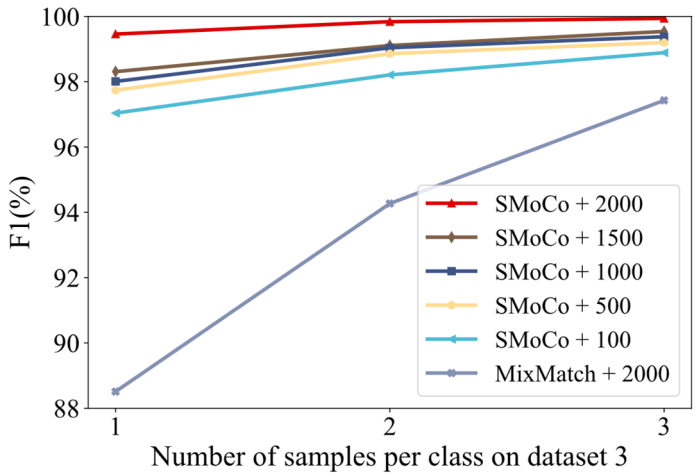



**Figure 18.** Experimental results of the sensitivity to the size of the data set on dataset 3.

From Table 8 and Figure 17, it can be seen that for dataset 2 when using 5 labeled training samples per class, all SMoCo with different unlabeled data sizes achieved excellent results. When using 1 labeled training sample per class, even SMoCo + 100 achieved a score of nearly 90%. SMoCo + 100 achieved a similar level of performance as with labeled pretraining and even reached the leading performance in the case of 1 sample per class and 5 samples per class, demonstrating the superior performance and robustness of SMoCo regarding the size of the unlabeled dataset. With the increase in data volume, SMoCo can achieve feature extractors with better performance via self-supervised learning.

As can be seen from Table 9 and Figure 18, SMoCo + 100 achieves excellent diagnostic performance even in the face of diagnostic problems across different devices and surprisingly greatly outperforms MixMatch + 2000. The progressive improvement in diagnostic

performance from SMoCo + 100 to SMoCo + 2000 proves that the performance of SMoCo can be increased gradually with the increase of the amount of data.

### *5.2. Sensitivity to Aero-Engine Bearing Dataset under Different Noise Levels*

In this section, noise stress tests are carried out to demonstrate the robustness and effectiveness of SMoCo with different signal-to-noise ratio (SNR) values on dataset 3. As a comparison, MixMatch and FFT + SVM are also used to perform diagnosis on dataset 3 with 3 samples per class at different noise levels, which are the best performing methods except SMoCo. SMoCo and MixMatch both use the full unlabeled dataset 1, i.e., 2000 samples per class. In this paper, we also further increase the difficulty of the experiment by training SMoCo from 1 sample per class to 3 samples per class of the labeled datasets, to verify the robustness of SMoCo under severe conditions, which are denoted as SMoCo + 1, SMoCo + 2, and SMoCo + 3. The evaluation criterion is the F1 score, and the results are shown in Table 10 and Figure 19.

**Table 10.** Experimental results of the sensitivity to the SNR on the aero-engine bearing dataset.

| Method | SNR | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | **0 dB** | **1 dB** | **2 dB** | **3 dB** | **4 dB** | **5 dB** | **6 dB** | **7 dB** | **8 dB** | **9 dB** | **10 dB** |
| SMoCo + 3 | 96.61 | 97.60 | 97.90 | 98.03 | 98.65 | 98.74 | 99.23 | 99.53 | 99.56 | 99.69 | 99.75 |
| SMoCo + 2 | 95.50 | 95.61 | 96.64 | 97.63 | 98.00 | 98.15 | 98.43 | 98.98 | 99.10 | 99.29 | 99.35 |
| SMoCo + 1 | 91.54 | 92.03 | 92.74 | 93.75 | 94.92 | 96.06 | 96.68 | 97.08 | 97.32 | 97.93 | 98.58 |
| MixMatch | 54.84 | 67.54 | 72.41 | 79.27 | 85.79 | 90.95 | 92.94 | 93.56 | 93.72 | 94.99 | 95.58 |
| FFT + SVM | 83.28 | 85.46 | 87.15 | 88.55 | 89.98 | 90.85 | 91.41 | 91.95 | 92.89 | 93.11 | 94.09 |

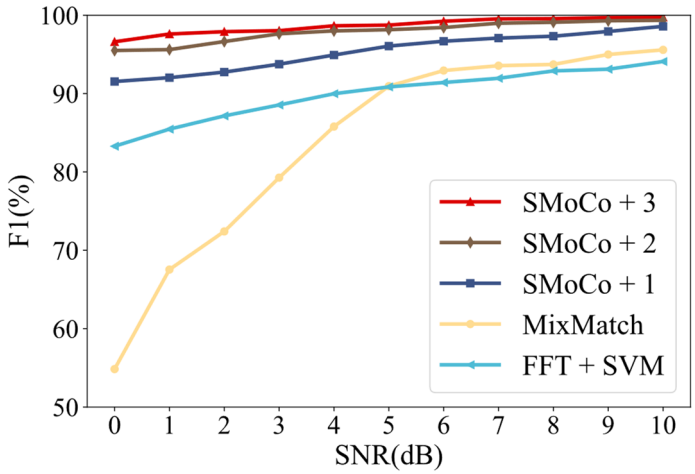



**Figure 19.** Experimental results of the sensitivity to the SNR on the aero-engine bearing dataset.

From Table 10 and Figure 19, it can be seen that SMoCo achieves the best result compared to the other two methods, even SMoCo + 1 can achieve a score of 91.54 at 0 dB, showing its strong robustness against noise. Although SMoCo + 1 achieves good diagnostic accuracy, the gap between it and SMoCo + 2 is large compared to the gap between SMoCo + 2 and SMoCo + 3, which is especially obvious in the case of strong noise. This is because, in the case of extremely small samples, there is a deviation in the decision boundary between the training and testing data sets due to interference of noise. In addition, MixMatch performs worse than FFT + SVM in the case of higher noise due to the fact that the gap between its data distribution and that of the unlabeled dataset gradually widens when the noise of the target diagnostic data increases, resulting in MixMatch not being able to make good use of the unlabeled data to improve the diagnostic accuracy of the target labeled data.

## 6. Conclusions

Under complex and harsh actual working conditions, there is a limited data problem in the fault diagnosis of aero-engine rolling bearings, which seriously affects the performance of intelligent diagnosis methods. Based on MoCo, this paper proposes a new intelligent diagnosis method based on SMoCo through improvement of the structure and innovation of the data augmentation method. SMoCo first performs self-supervised learning on easily available unlabeled data and then utilizes the trained feature extractor for downstream diagnostic tasks under limited data. Experimental results show that SMoCo not only has high diagnostic accuracy and training efficiency, but also has good generalization ability. The experimental results show that SMoCo can have high diagnostic accuracy and training efficiency under limited data, whether the target data are from the same model but with different failure modes and different working conditions or from a completely different type from the pre-training data, which proves its good generalization ability. The main conclusions are as follows:

1. In this paper, BN and a predictor are introduced to solve the deficiency of the MoCo structure, and SML is innovatively proposed according to the time domain and frequency domain of the signal, which regards the time-domain signal and frequency-domain signal as a positive pair. Therefore, a fault diagnosis method based on SMoCo is proposed.
2. SMoCo uses easily available unlabeled data for self-supervised learning, the sources of which can be diverse and are not limited to objects that need to be diagnosed. Therefore, its acquisition range is wider, and its feasibility in practical diagnostic tasks is much greater than that of previous work.
3. This paper uses two independent bearing datasets from Paderborn University and the Polytechnic University of Turin for experimental verification. In the experiment, three important problems of aero-engine bearing fault diagnosis under the condition of limited data are studied, which are different working conditions, different failure modes, and different equipment. After SMoCo performs self-supervised learning on artificially injected faulted bearings, the trained feature extractor can be used to solve the above problems. The results show that the proposed SMoCo method can effectively solve the diagnosis problem in the case of limited data, it greatly exceeds the existing state-of-the-art methods both in accuracy and speed and is very little affected by limited data, even requiring only one sample per class to achieve high diagnostic accuracy for aero-engine bearing.
4. Compared with representative methods, SMoCo still achieves good performance in the case of limited unlabeled pre-training data and less labeled training data with strong noise, demonstrating the robustness of SMoCo regarding data volume and noise.

Although the SMoCo proposed in this paper has achieved good results, there is still some work that deserves further exploration, especially in relation to the time and efficiency of self-supervised learning. SMoCo takes a relatively long time to learn the essential features of the signal in the self-supervised learning phase, and future research could be conducted to improve the training efficiency. In addition, in this paper, only Gaussian noise is explored as the data augmentation method, while there are often other non-Gaussian noises and mixed noises in the actual industry [34,35], which could be further investigated in the future to be more robust regarding the complex conditions in actual industry. Future work could also try to change the structure of the encoder to use a convolutional network with better performance or a transformer network, which is currently performing extremely well in the field of deep learning [36]. A larger pre-trained dataset with different data sources, not just from one bearing dataset, could be used to try to build a unified feature extractor for all rotating machinery problems.

**Author Contributions:** Conceptualization, Z.Y.; methodology, Z.Y. and H.L.; software, Z.Y.; validation, Z.Y.; writing—original draft, Z.Y.; writing—review and editing, H.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This study was supported by the National Natural Science Foundation of China (Grant No. 61973011).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

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

## References

1. Wang, B.; Zhang, X.; Sun, C.; Chen, X. A Quantitative Intelligent Diagnosis Method for Early Weak Faults of Aviation High-Speed Bearings. *ISA Trans.* **2019**, *93*, 370–383. [CrossRef] [PubMed]
2. Jiang, X.; Huang, Q.; Shen, C.; Wang, Q.; Xu, K.; Liu, J.; Shi, J.; Zhu, Z. Synchronous Chirp Mode Extraction: A Promising Tool for Fault Diagnosis of Rolling Element Bearings under Varying Speed Conditions. *Chin. J. Aeronaut.* **2022**, *35*, 348–364. [CrossRef]
3. Wang, Y.; Tse, P.W.; Tang, B.; Qin, Y.; Deng, L.; Huang, T. Kurtogram Manifold Learning and Its Application to Rolling Bearing Weak Signal Detection. *Measurement* **2018**, *127*, 533–545. [CrossRef]
4. Lei, Y.; Yang, B.; Jiang, X.; Jia, F.; Li, N.; Nandi, A.K. Applications of Machine Learning to Machine Fault Diagnosis: A Review and Roadmap. *Mech. Syst. Signal Process.* **2020**, *138*, 106587. [CrossRef]
5. Feng, Y.; Chen, J.; Zhang, T.; He, S.; Xu, E.; Zhou, Z. Semi-Supervised Meta-Learning Networks with Squeeze-and-Excitation Attention for Few-Shot Fault Diagnosis. *ISA Trans.* **2022**, *120*, 383–401. [CrossRef] [PubMed]
6. Yu, K.; Lin, T.R.; Ma, H.; Li, X.; Li, X. A Multi-Stage Semi-Supervised Learning Approach for Intelligent Fault Diagnosis of Rolling Bearing Using Data Augmentation and Metric Learning. *Mech. Syst. Signal Process.* **2021**, *146*, 107043. [CrossRef]
7. Zhang, S.; Ye, F.; Wang, B.; Habetler, T.G. Semi-Supervised Bearing Fault Diagnosis and Classification Using Variational Autoencoder-Based Deep Generative Models. *IEEE Sens. J.* **2021**, *21*, 6476–6486. [CrossRef]
8. Wen, L.; Gao, L.; Li, X. A New Deep Transfer Learning Based on Sparse Auto-Encoder for Fault Diagnosis. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 136–144. [CrossRef]
9. Wang, Y.; Sun, X.; Li, J.; Yang, Y. Intelligent Fault Diagnosis With Deep Adversarial Domain Adaptation. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 3035385. [CrossRef]
10. Zheng, H.; Wang, R.; Yang, Y.; Yin, J.; Li, Y.; Li, Y.; Xu, M. Cross-Domain Fault Diagnosis Using Knowledge Transfer Strategy: A Review. *IEEE Access* **2019**, *7*, 129260–129290. [CrossRef]
11. Jing, L.; Tian, Y. Self-Supervised Visual Feature Learning With Deep Neural Networks: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 4037–4058. [CrossRef]
12. Zhang, R.; Isola, P.; Efros, A.A. Colorful Image Colorization. In *Computer Vision–ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2016; Volume 9907, pp. 649–666; ISBN 978-3-319-46486-2.
13. Pathak, D.; Krähenbühl, P.; Donahue, J.; Darrell, T.; Efros, A.A. Context Encoders: Feature Learning by Inpainting. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2536–2544.
14. Noroozi, M.; Favaro, P. Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles. In *Computer Vision–ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 69–84.
15. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 105–114.
16. Wang, H.; Liu, Z.; Ge, Y.; Peng, D. Self-Supervised Signal Representation Learning for Machinery Fault Diagnosis under Limited Annotation Data. *Knowl. Based Syst.* **2022**, *239*, 107978. [CrossRef]
17. Tian, Y.; Sun, C.; Poole, B.; Krishnan, D.; Schmid, C.; Isola, P. What Makes for Good Views for Contrastive Learning? In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2020; Volume 33, pp. 6827–6839.
18. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. In Proceedings of the 37th International Conference on Machine Learning, PMLR, Vienna, Austria, 21 November 2020; pp. 1597–1607.
19. Chen, T.; Kornblith, S.; Swersky, K.; Norouzi, M.; Hinton, G.E. Big Self-Supervised Models Are Strong Semi-Supervised Learners. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2020; Volume 33, pp. 22243–22255.
20. Caron, M.; Misra, I.; Mairal, J.; Goyal, P.; Bojanowski, P.; Joulin, A. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2020; Volume 33, pp. 9912–9924.

21. Grill, J.-B.; Strub, F.; Altché, F.; Tallec, C.; Richemond, P.H.; Buchatskaya, E.; Doersch, C.; Pires, B.A.; Guo, Z.D.; Azar, M.G.; et al. Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning. *Adv. Neural Inf. Processing Syst.* **2020**, *33*, 21271–21284.
22. Wei, M.; Liu, Y.; Zhang, T.; Wang, Z.; Zhu, J. Fault Diagnosis of Rotating Machinery Based on Improved Self-Supervised Learning Method and Very Few Labeled Samples. *Sensors* **2021**, *22*, 192. [CrossRef]
23. Ding, Y.; Zhuang, J.; Ding, P.; Jia, M. Self-Supervised Pretraining via Contrast Learning for Intelligent Incipient Fault Detection of Bearings. *Reliab. Eng. Syst. Saf.* **2022**, *218*, 108126. [CrossRef]
24. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum Contrast for Unsupervised Visual Representation Learning. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.
25. Peng, T.; Shen, C.; Sun, S.; Wang, D. Fault Feature Extractor Based on Bootstrap Your Own Latent and Data Augmentation Algorithm for Unlabeled Vibration Signals. *IEEE Trans. Ind. Electron.* **2022**, *69*, 9547–9555. [CrossRef]
26. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* **2015**, arXiv:1502.03167.
27. Oord, A.; van den Li, Y.; Vinyals, O. Representation Learning with Contrastive Predictive Coding. *arXiv* **2018**, arXiv:1807.03748.
28. Lessmeier, C.; Kimotho, J.K.; Zimmer, D.; Sextro, W. Condition Monitoring of Bearing Damage in Electromechanical Drive Systems by Using Motor Current Signals of Electric Motors: A Benchmark Data Set for Data-Driven Classification. In Proceedings of the PHM Society European Conference, Bilbao, Spain, 5–8 July 2016; Volume 3. [CrossRef]
29. Zhao, Z.; Li, T.; Wu, J.; Sun, C.; Wang, S.; Yan, R.; Chen, X. Deep Learning Algorithms for Rotating Machinery Intelligent Diagnosis: An Open Source Benchmark Study. *ISA Trans.* **2020**, *107*, 224–255. [CrossRef]
30. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
31. Berthelot, D.; Carlini, N.; Goodfellow, I.; Papernot, N.; Oliver, A.; Raffel, C.A. MixMatch: A Holistic Approach to Semi-Supervised Learning. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: New York, NY, USA, 2019; Volume 32.
32. Daga, A.P.; Fasana, A.; Marchesiello, S.; Garibaldi, L. The Politecnico Di Torino Rolling Bearing Test Rig: Description and Analysis of Open Access Data. *Mech. Syst. Signal Process.* **2019**, *120*, 252–273. [CrossRef]
33. Forouzanfar, M.; Safaeipour, H.; Casavola, A. Oscillatory Failure Case Detection in Flight Control Systems via Wavelets Decomposition. In *ISA Transactions*; Elsevier: Amsterdam, The Netherlands, 2021. [CrossRef]
34. Safaeipour, H.; Forouzanfar, M.; Ramezani, A. Incipient Fault Detection in Nonlinear Non-Gaussian Noisy Environment. *Measurement* **2021**, *174*, 109008. [CrossRef]
35. Ortiz Ortiz, F.J.; Rodríguez-Ramos, A.; Llanes-Santiago, O. A Robust Fault Diagnosis Method in Presence of Noise and Missing Information for Industrial Plants. In *Proceedings of the Pattern Recognition*; Springer International Publishing: Cham, Switzerland, 2022; pp. 35–45.
36. Fang, H.; Deng, J.; Bai, Y.; Feng, B.; Li, S.; Shao, S.; Chen, D. CLFormer: A Lightweight Transformer Based on Convolutional Embedding and Linear Self-Attention With Strong Robustness for Bearing Fault Diagnosis Under Limited Sample Conditions. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 3132327. [CrossRef]