

Article

Few-Shot Learning for Fault Diagnosis: Semi-Supervised Prototypical Network with Pseudo-Labels

Jun He ¹ , Zheshuai Zhu ^{1,*}, Xinyu Fan ², Yong Chen ¹, Shiya Liu ¹ and Danfeng Chen ¹

¹ The College of Mechanical Engineering and Automation, Foshan University, Foshan 528011, China; hejun_723@fosu.edu.cn (J.H.); chen@fosu.edu.cn (Y.C.); liushiya@fosu.edu.cn (S.L.); cdf2017@fosu.edu.cn (D.C.)

² The School of Automation, Central South University, Changsha410017, China; auxyfan@csu.edu.cn

* Correspondence: 2112052078@stu.fosu.edu.cn; Tel.: +86-188-6809-2089

Abstract: Achieving deep learning-based bearing fault diagnosis heavily relies on large labeled training samples. However, in real industry applications, labeled data are scarce or even impossible to obtain. In this study, we addressed a challenging few-shot bearing fault diagnosis problem with few or no training labeled samples of novel categories. To tackle this problem, we considered a semi-supervised prototype network based on few-shot bearing fault diagnosis with pseudo-labels. The existing prototypical networks with pseudo-label methods train a pseudo label model to label unlabeled samples using high-dimensional labeled data, which cannot eliminate the instability of the pseudo-label model caused by dimensional labeled features. To mitigate this issue, we used kernel principal component analysis to reduce the dimensions of and remove redundant information from high-dimensional data. Specifically, we used the pseudo-label prediction algorithm with probability distance to label unlabeled samples, aiming to improve the labeling accuracy. We applied two well-known bearing data sets for the validation experiments with symmetry parameters. The findings illustrated that the classification accuracy of the proposed method is higher than that of other existing methods.



Citation: He, J.; Zhu, Z.; Fan, X.; Chen, Y.; Liu, S.; Chen, D. Few-Shot Learning for Fault Diagnosis: Semi-Supervised Prototypical Network with Pseudo-Labels. *Symmetry* **2022**, *14*, 1489. <https://doi.org/10.3390/sym14071489>

Academic Editor: Jan Awrejcewicz

Received: 30 June 2022

Accepted: 19 July 2022

Published: 21 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: few-shot learning; prototype networks; multi-kernel PCA; pseudo label

1. Introduction

Rotating machinery is an important component of smart manufactory, and its healthy and stability operation are required to guarantee production. However, due to bearings operating long term in harsh environments, then can easily fail, leading to disastrous consequences [1,2]. To ensure the safety and efficiency of smart manufacturing, rolling bearing fault diagnosis needs to be further studied, which has increasingly attracted research attention [3,4].

Benefitting from the rapid development of computer and sensing technologies, industry has entered the era of big data. Due to its big data learning ability, deep learning has replaced shallow models and has been successfully applied in various fields [5–7]. With continuous development and improvement, deep-learning models have been widely applied in the field of fault diagnosis. Gong et al. [8] use an improved convolutional neural network support vector machine (CNN-SVM) method to effectively identify incipient faults in rotation machinery. Jiang et al. [9] explored a deep recurrent neural network (DRNN) to automatically extract the features from input spectrum sequences. Cui et al. [10] proposed a feature distance stack autoencoder (FD-SAE) for rolling bearing fault diagnosis to improve the feature extraction ability and the convergence speed of the network. Zhang et al. [11] combined an ensemble deep belief network and variation mode decomposition to improve the accuracy and stability of the diagnosis of the health status of rotating machinery. These existing methods, which are based on deep learning, can produce accurate results; however, a large amount of labeled data are required to obtain an effective deep model. However,

obtaining enough labeled samples in actual industrial applications is difficult and time-consuming. Therefore, more effort is needed to apply few-shot learning to solving the problems encountered in the practical application of deep-learning methods.

Few-shot learning, a technique of learning from a few labeled samples for automatically classifying massive amounts unlabeled samples, has recently attracted attention. Zhang et al. [12] proposed a few-shot learning approach for fault diagnosis under limited data conditions based on CNNs and Siamese neural networks. Jiang et al. [13] embedded a two-branch network into the prototype network, building a two-branch prototype network fault diagnosis method to mitigate the few-shot samples classification issue. Xu et al. [14] combined K-nearest neighbor with cosine distance to build a distribution discrepancy metric and developed a deep convolutional nearest neighbor matching network for few-shot learning. Wang et al. [15] developed a feature space metric-based meta-learning model to overcome the challenge produced by few-shot learning problem under limited labeled samples by adopting both individual sample information and similarity sample group information. Xu et al. [16], based on approximation space and belief functions, designed a few-shot learning method for fault diagnosis. They used the basic probability assignment calculation to build belief functions for diagnosis within sufficient information. Although these existing few-shot learning fault diagnosis methods achieve encouraging fault diagnosis performance in the conditions with few labeled samples, these existing few-shot methods ignore the massive unlabeled data that exist in practical industrial applications, which may be used as pseudo-labels in combination with the few labeled samples to effectively train deep-learning models.

Semi-supervised few-shot learning, as a promising method for labeled samples, is increasingly receiving research attention. Tao et al. [17] designed a bearing defect diagnosis model using pseudo-labels, which obtained representative features for classification from unlabeled samples. Zhang et al. [18] used a Monte Carlo uncertainty threshold selection strategy to increase the confidence of the pseudo-labels, then used a momentum prototype network to obtain the feature space mapping using few labeled samples. Yong et al. [19] used an encoder to extract features for training prototypes, then semi-supervised meta-learning, which they optimized by a combinatorial learning optimizer to refine original prototypes from unlabeled samples. Kai et al. [20] explored a pseudo-loss confidence metric for task-unified confidence estimation through mapping the different pseudo-labels to the same metric space using of the pseudo-loss. Di et al. [21] combined learner of latent representations with cluster structures, and proposed a pseudo-label-guided collective matrix factorization method for multi-view clustering. More recently, these semi-supervised few-shot learning methods have been widely studied, and encouraging results have been obtained. However, most of these existing methods focus on the confidence estimation inference of pseudo-label learning, which suffers when samples in a single task are insufficient, and ignore the redundant information embedded into feature space, which can cause errors in pseudo-label learning models, considerably decreasing the generalization capability of a model.

In an effort to achieve the semi-supervised few-shot learning, and motivated by the aforementioned analysis, and considering the influence of the redundant information in high-dimensional feature space, we designed a kernel principal component analysis method based on a semi-supervised prototypical network for fault diagnosis with pseudo-labels. We used kernel principal component analysis to reduce the dimensions of the feature space, which mitigates the effects of redundant information and results in a lightweight training model. We used the pretrained model, whose parameters we obtained by training model with these features through dimension reduction, to predict the labels of unlabeled data, and we selected the reliable labels as the pseudo-labels. We sent the pseudo-labeled data to the pretrained prototype networks to further fine tune the parameters to produce prototype networks with strong generalization ability. Finally, we conducted comparison experiments based on two well-known bearing datasets (Case Western Reserve University, CWRU) to prove the effectiveness of our method.

The main highlights of the study are as follows:

- (1) We used kernel principal component analysis to reduce the dimension of the feature space, which avoid redundant information embedded in the feature space reducing the generalization ability of the model;
- (2) We used apseudo-label-prediction algorithm to generate labeled samples, aiming to increase the labeled samples, which fully uses the unlabeled samples for training the prototype networks to avoid overfitting;
- (3) We adopted predicted pseudo-label data to fine tune the prototype network parameters, which can reduce the time required for adjusting the model parameters and improves diagnostic accuracy.

2. Theoretical Background

2.1. Few-Shot Learning

Few-shot learning [22], which aims to learn about a new category from a small amount of labeled data, has aroused increased interest in the pattern recognition community. This technique has been extensively applied in artificial intelligence.

In the few-shot learning problem, all datasets are divided into three parts: the support set $S = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ ($n_s = C \times N$, $x_i \in R^D$), query set $Q = \{(x_i^q, y_i^q)\}_{i=1}^{n_q}$ ($n_q = 1 \times N$, $x_i \in R^D$), and test set $T = \{(x_i^t, y_i^t)\}_{i=1}^{n_t}$, where the data of support and query sets are from the same class, and the samples in the two sets are different. The sample categories of the test set differ from those of the support set. $x_i \in R^D$ denote the feature vector extracted from a raw vibration signal; $y_i \in \{1, 2, \dots, C\}$ denotes the label of the dataset samples. In the traditional method, the dataset is only divided into a training and a test set. Compared with the traditional method, the support set S and query set Q are used to train the network, and the test set T is used to evaluate the performance of the network, which improves the stability and generalization of the model. If the support set contains N classes and K samples, it can be described as an N -way K -shot problem. The process of few-shot learning is illustrated in Figure 1.

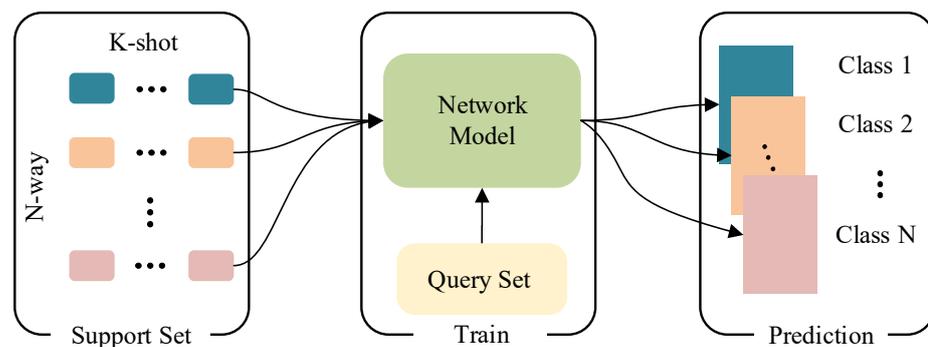


Figure 1. N-way K-shot learning problem description.

2.2. Prototypical Network

Prototypical networks [23,24] generalize new classes not included in the training set given only a small number of examples of each new class. Metric-based few-shot learning has been widely used in few-shot learning, producing impressive results. An embedding function is obtained by a neural network through prototypical network learning; then, samples are extracted into feature vectors. The mean vectors in each class are the prototype. During classification, query samples are first transformed into feature vectors; the distance from the vector to the prototypes represents the similarity to the class. Figure 2 describes the working of the prototype networks.

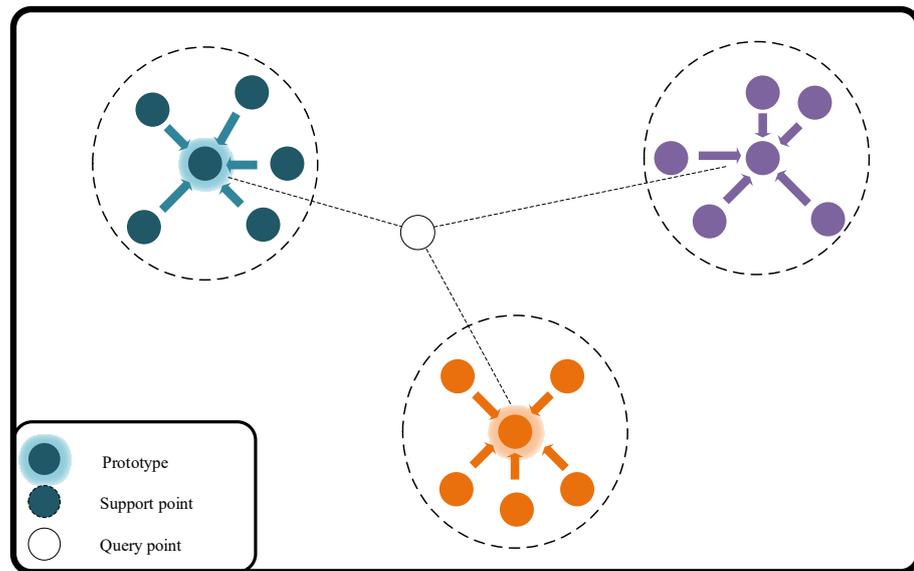


Figure 2. Schematic diagram of prototype network.

In few-shot learning, a support set of N labeled samples $S = \{(x_1, y_1), \dots, (x_n, y_n)\}$ is given, where each $x_i \in R^D$ is the dimensional feature vector D , and $y_i \in \{1, \dots, K\}$ is the label of x_i . S_k denotes k classes in the support set. Through an embedding function $f_\phi: R^D \rightarrow R^M$, prototype p_k is computed as follows:

$$p_k = \frac{1}{|S_k|} \sum_{(x_i, y_i) \in S_k} f_\phi(x_i) \quad (1)$$

During the classification, distance is calculated by the distance function $d(\cdot)$; the probability that query point X belongs to class K can be expressed as:

$$P_\phi(y = k|x) = \log \left(\frac{\exp(-d(f_\phi(x), p_k))}{\sum_{k'} \exp(-d(f_\phi(x), p_{k'}))} \right) \quad (2)$$

3. Proposed Method

3.1. KPAC

KPCA is a non-linear derivative of PCA that can be solved as an eigenvalue problem of its kernel matrix [25]. Samples are non-linearly mapped into higher-dimensional feature space F , and PCA is performed there.

Let sample $x_1, \dots, x_N \in R^D$ be mapped into $\phi(x_1), \dots, \phi(x_n) \in F$. The covariance matrix C in the feature space F is given by:

$$C = \frac{1}{N} \sum_{k=1}^N \phi(x_k) \phi(x_k)^T \quad (3)$$

where $\sum_{k=1}^N \phi(x_k) = 0$. Non-zero eigenvalues λ of the covariance matrix C can be calculated as:

$$\lambda v = C v \quad (4)$$

where v denotes the corresponding eigenvector of F , which can also be written as:

$$v = \sum_{k=1}^N \alpha_k \phi(x_k) \quad (5)$$

The problem is simplified to find the coefficient l_k , which can be formulated as the following eigenvalue problem by substituting Equations (3) and (5) into (4), which is written as:

$$K\alpha = N\lambda\alpha \quad (6)$$

where K is a kernel matrix with size $N \times N$, which can be calculated as follows:

$$K_{ij} = (\phi(x_i) \cdot \phi(x_j)) = k(x_i, x_j) \quad (7)$$

$k(\cdot)$ denotes the kernel function, which is used to calculate the inner product of $\phi(x_i)$ and $\phi(x_j)$. In this study, we used the RBF kernel function:

$$k(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (8)$$

where σ is set to $1/N$. Let λ_l be the l th largest eigenvalue of K , and $\alpha_l = [\alpha_1^l, \dots, \alpha_N^l]$ be the corresponding eigenvector. An input sample x can be mapped onto the l th dimension of KPCA space with coordinate value:

$$\langle v_l, \phi(x) \rangle = \sum_{i=1}^N \alpha_i^l k(x_i, x) \quad (9)$$

The advantage of kernel principal component analysis is that only the kernel function needs to be calculated in the original space; the nonlinear mapping function $\phi(x)$ does not need to be known.

3.2. Metric and Query

In this study, we used the Euclidean distance to calculate the similarity of samples through the feature vectors extracted from these samples. The distance can be represented as:

$$d_f(p_n, x_i^q) = \sqrt{\|f(p_n) - f(x_i^q)\|} \quad (10)$$

where p_n denotes the prototype of class n , x_i^q denotes the i th sample in query set, and $f(\cdot)$ is the feature vector extracted from the raw vibration signal. The smaller the distance d_f , the more similar the query data to this class. The probability of the sample from query set x_i^q belonging to class k can be described as:

$$P(y = k | x_i^q) = \frac{\exp(-d(f(x), p_n))}{\sum_n \exp(-d(f(x), p_n))} \quad (11)$$

In the process of pseudo-label learning, to retain samples with good classification performance, we used the experimental data to verify that, after SoftMax function screening, the samples whose probability value was greater than 0.7 had high classification accuracy. The detailed experiment of probability P is illustrated in Figure 3.

Then, the loss function of the samples selected for the query set was designed as:

$$\ell_{loss} = -\log P(y = k | x_i^q) \quad (12)$$

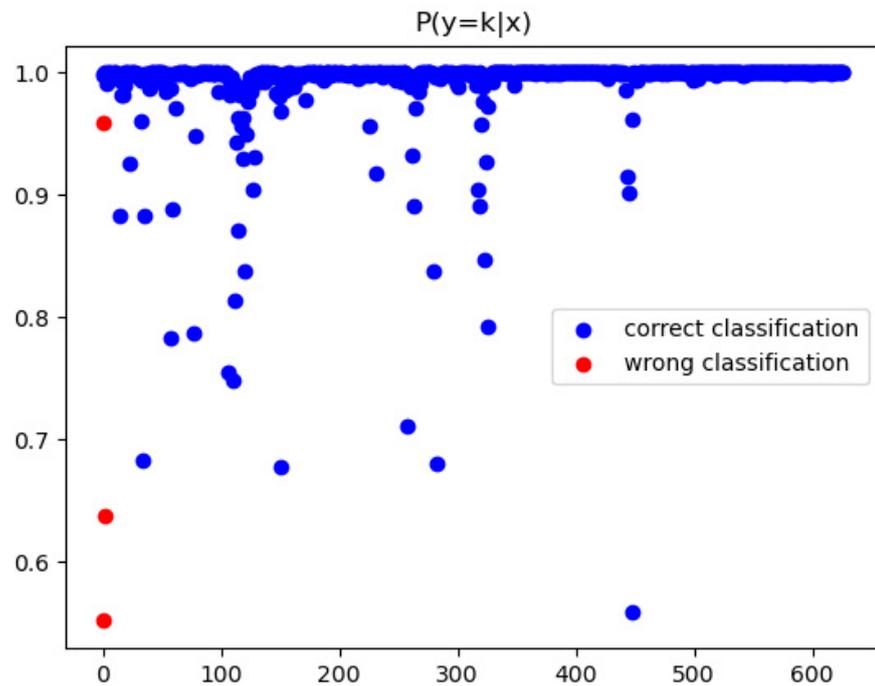


Figure 3. Probability P vs. the classification accuracy.

3.3. Description of Proposed Method

In this study, we designed a kernel principal analysis based semi-supervised prototypical network (PSSPN). The whole algorithm is described in Algorithm 1.

Algorithm 1: PSSPNlearning strategy

Input: Labeled dataset D_L , unlabeled dataset D_U , number of fault classes N , support set S_e with K samples query set Q_e with Q samples, feature extractor f_φ episode, and epoch.

Output: learnable parameter φ

1. Preprocess the raw data with KPCA
 2. For each epoch, do:
 3. Randomly sample N classes in dataset D_L ; each class has K samples THAT consist of support set S_e . Similarly, randomly sample Q samples to create query set Q_e .
 4. Obtain samples x_i^S and x_i^Q from support set S_e and query set Q_e , respectively, and generate support feature set $f_\varphi(x_i^S)$ and query feature set $f_\varphi(x_i^Q)$.
 5. Generate prototype P_k by Equation (1).
 6. Calculate the classification probability $P(y = k | x_i^Q)$ by Equation (2).
 7. Calculate the loss, and update parameter φ .
 8. Use the model pretrained in steps 2–8 to predict the label of D_U ; after selection, we obtain the pseudo-labeled dataset D_{pseudo} .
 9. Fine-tune parameter φ with datasets D_{pseudo} and D_L .
 10. End
-

Step (1)—Few-shot learning: the dimensions of the labeled data feature space are reduced by KPCA. Then, feed this reduced-dimension feature space into the prototypical network. Calculate the distance between samples in the query set and prototype. Then, convert the similar distances into probability values using a SoftMax classifier. Calculate the loss and update the parameters of the network. After iterating, the pretrained model is obtained.

Step (2)—Unlabeled samples data are preprocessed by KPCA: through the pretrained model, obtain the predicted label of the unlabeled data. After selection, retain part of the label, and then obtain the pseudo-labeled data.

Step (3)—Input the predicted pseudo-label samples to the labeled sample set to train and fine-tune the relevant parameters of the prototypical network.

A detailed description of the workflow of the proposed method is provided in Figure 4.

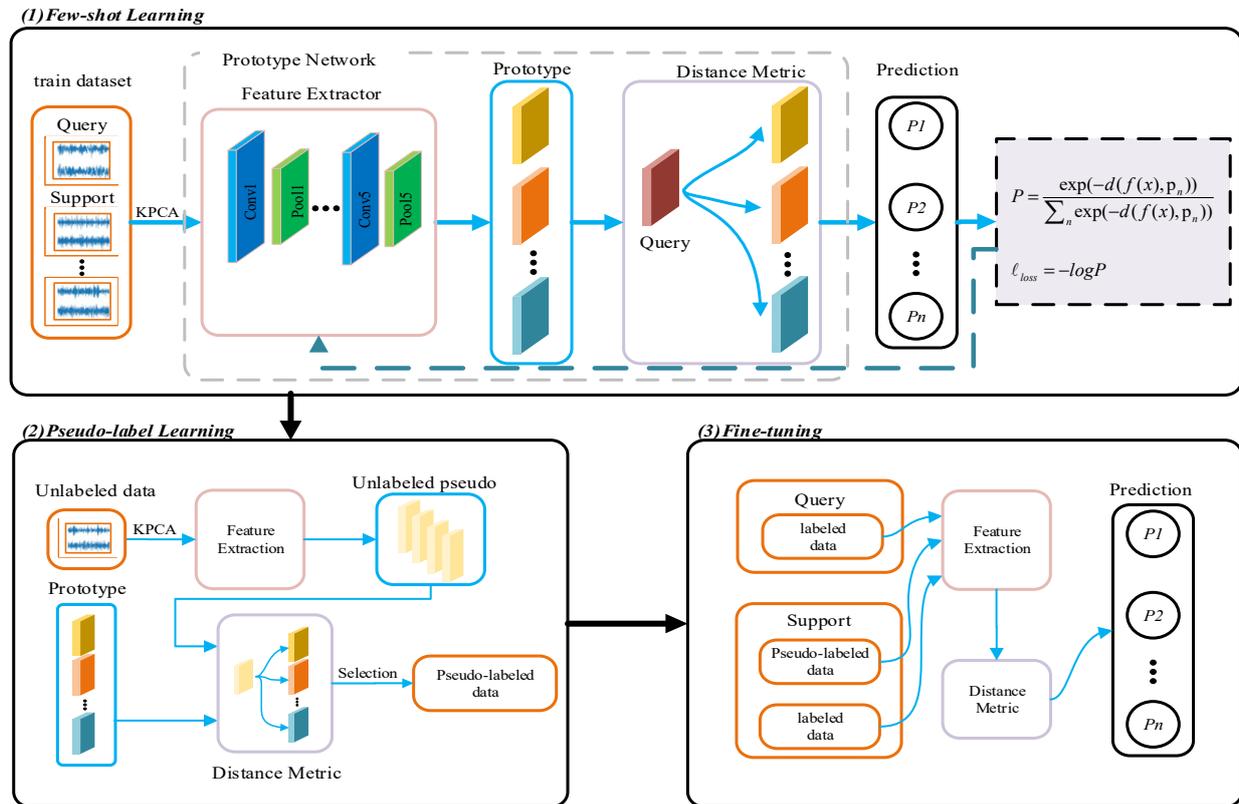


Figure 4. Fault diagnosis framework of the proposed method.

4. Results and Discussion

We used three methods, CNN, ProtoNet [24], and improved prototype network (IPN) [26], for a comparison experiment to verify the validity of the proposed method. The feature extractors of CNN and ProtoNet have the same network structure. IPN uses L2 regularization and a dropout layer, which help to address model overfitting problem. The architecture of PSSPN is described in detail in Table 1. We use leaky ReLU as the activation function, and α is 0.3. The network is optimized by the Adam optimizer, whose learning rate is 0.001. We repeated the experiment 20 times to obtain the final accuracy. We used Tensorflow 2.0 to conduct the experiment.

Table 1. Feature extractor network description.

No.	Layer Type	Kernel Size/Stride	Kernel Number	Output Size (Width × Depth)	Padding
1	Conv1	3 × 1/1 × 1	16	256 × 16	same
2	Pool1	2 × 1/1 × 1	16	128 × 16	valid
3	Conv2	3 × 1/1 × 1	32	128 × 32	same
4	Pool2	2 × 1/1 × 1	32	64 × 64	valid
5	Conv3	3 × 1/1 × 1	64	32 × 64	same
6	Pool3	2 × 1/1 × 1	64	16 × 64	valid
7	Conv4	3 × 1/1 × 1	64	16 × 64	same
8	Pool4	2 × 1/1 × 1	64	6 × 64	valid
9	Conv5	3 × 1/1 × 1	64	6 × 64	same
10	Pool5	2 × 1/1 × 1	64	3 × 64	valid

4.1. Case Study on CWRU

4.1.1. Description of CWRU

In this study, we used the Case Western Reserve University (CWRU) bearing datasets [27], which were collected under four different working conditions and loads (0, 1, 2, and 3 hp); the motor worked at speeds of 1979, 1772, 1750, and 1730 rpm, respectively. Each working condition had four bearing fault conditions: normal, ball fault, inner race fault, and outer race fault. Each fault type contained three fault sizes: 0.007, 0.014, and 0.021 inches. We provide details about the dataset in Table 2, which shows that there were 10 types in total. For the detailed experimental platform, please refer to the related references.

Table 2. Bearing health states in CWRU dataset.

State	Description	Fault Size (Inches)
N	Normal condition	
RF	Fault on roller	0.007, 0.014, 0.021
IF	Fault on inner race	0.007, 0.014, 0.021
OF	Fault on te out race	0.007, 0.014, 0.021

We generated all the training and testing samples using a sliding window. We set the sliding window to 1024, and step length of the framing to 80. The detailed information about the dataset used in the experiment is provided in Table 3.

Table 3. Information about dataset used in our experiments.

Fault Location	None	Ball			Inner Race			Outer Race			Load
Fault Diameter (Inches)	0	0.007	0.014	0.021	0.007	0.014	0.021	0.007	0.014	0.024	
Fault Labels	1	2	3	4	5	6	7	8	9	10	
Pretrain	500	500	500	500	500	500	500	500	500	500	
Dataset Unlabeled	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1
Test	200	200	200	200	200	200	200	200	200	200	

4.1.2. Results Analysis

In this case, study, 1-shot and 5-shot experiment is conducted on the dataset described as Table 3, all parameters of compared methods are mentioned above, and the experiment result is listed in Table 4

Table 4. The classification accuracy of methods based on CWRU.

Methods	CWRU				
	1-Shot	5-Shot	10-Shot	15-Shot	30-Shot
CNN	18.88 ± 0.23	80.58 ± 0.38	80.36 ± 0.54	80.09 ± 0.07	80.32 ± 0.17
ProtoNet [24]	83.06 ± 0.77	89.80 ± 0.32	92.27 ± 0.23	99.81 ± 0.02	99.85 ± 0.01
IPN [28]	85.97 ± 0.43	89.97 ± 0.23	93.36 ± 0.31	99.59 ± 0.03	99.28 ± 0.05
ProtoNet+KPCA	85.12 ± 0.58	92.03 ± 0.23	95.88 ± 0.10	96.30 ± 0.13	96.26 ± 0.09
PSSPN (ours)	89.72 ± 0.38	94.65 ± 0.16	97.05 ± 0.07	95.92 ± 0.10	96.59 ± 0.07

Table 4 shows that the classification accuracies of the few-shot learning methods are much higher than that of conventional CNN. For 1-, 5-, and 10-shot learning, the PSSPN achieved 89.72%, 94.65%, and 97.05% accuracies, respectively, which are higher than those of the other considered methods. The classification accuracy of ProtoNet with KPCA is higher than that of ProtoNet. This finding showed that KPCA helps remove redundant information, which cause errors in the model during training.

Figure 5 shows the confusion matrix for the five-shot classification accuracy result of PSSPN. We found that the We generated accuracy of PSSPN was high for various bearing fault types. However, for label 8, the classification accuracy was relatively low, and several samples were categorized into label 9. The most probable reason for this is that the difference between these two samples is small, which lead to misclassification. With the increase in the number of training samples, the classification accuracy of the various methods also increased, especially that of ProtoNet and IPN. The classification accuracy of PSSPN was higher when the number of labeled training samples was small.

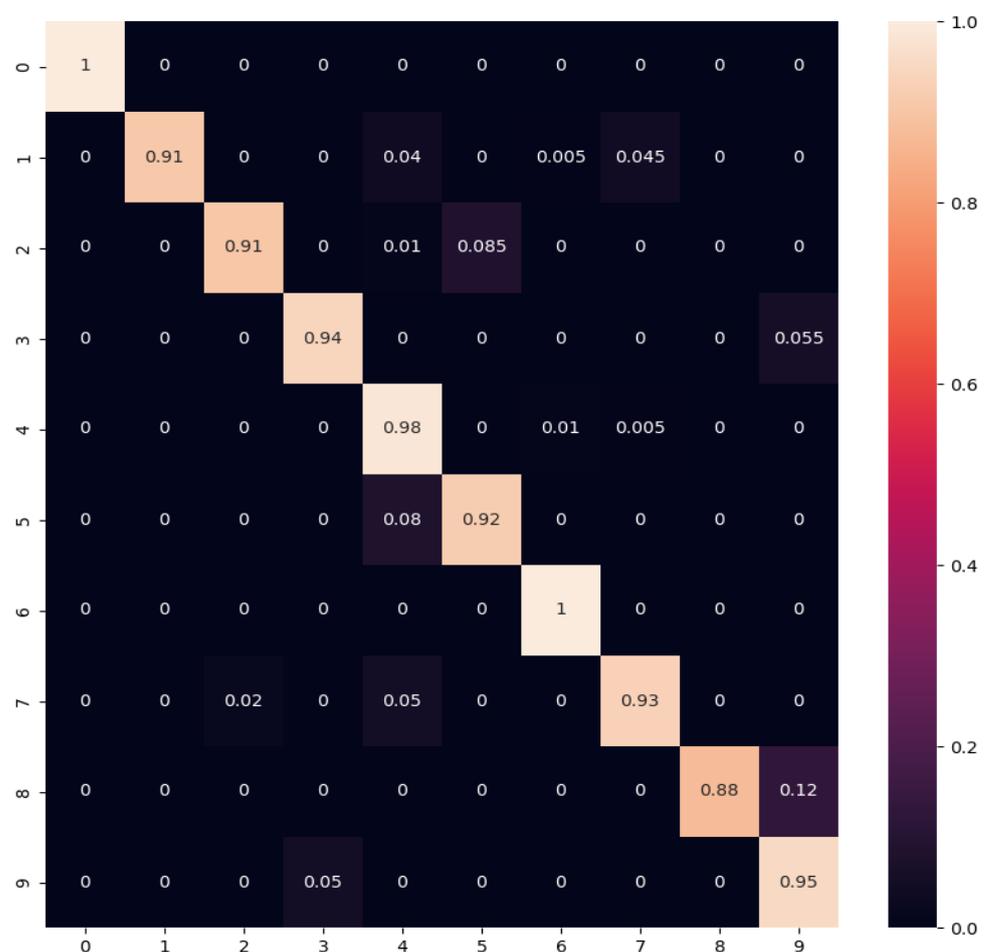
**Figure 5.** Confusion matrix for 5-shot classification result produced by PSSPN in this case study.

Figure 6 illustrates the feature visualization produced for t-SNE. In CNN, the difference in some classes was clear, but several features were indivisible. In ProtoNet and IPN, different samples were successfully distinguished, but the result was messy. The results produced by PSSPN were more clearly divided than that of the other considered methods, though the boundaries of some samples were ambiguous. Possible reasons for this include the amount of data being relatively small, and the model could not be further improved; or the difference between the samples being small, and our model could not distinguish this gap.

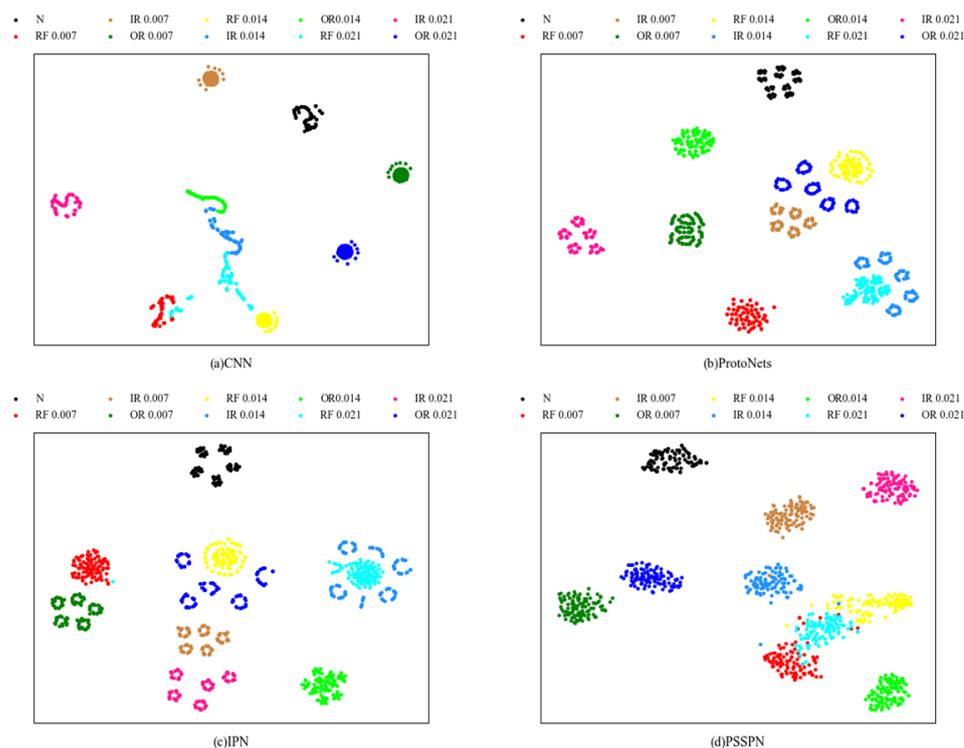


Figure 6. t-SNE visualization produced by different methods in this case study.

4.2. Case Study on Petrochemical Dataset

4.2.1. Petrochemical Dataset Introduction

The petrochemical dataset (Guangdong Provincial Key Laboratory of Petrochemical Equipment Fault Diagnosis, Guangdong University of Petrochemical Technology, Maoming, China) [29,30] contains more noise and is related to industrial environments. We established a simulation platform to simulate the actual working environment of a petrochemical refinery and the power load of rotating machinery. The detailed information of the platform used for data collection from the machinery is provided in Figure 7. For more detailed information, please refer to the provided references.

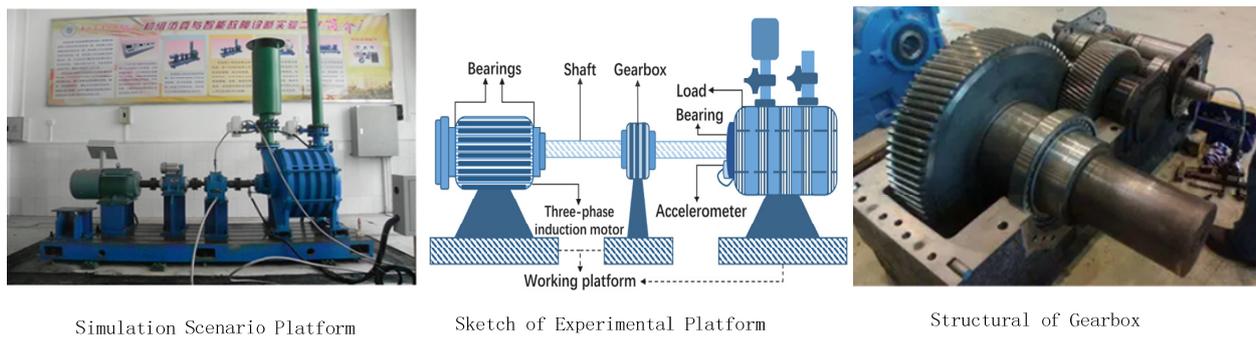


Figure 7. Platform of petrochemical motor.

The fault types in the petrochemical dataset are as follows: (1) F0: gearwheels are missing teeth; (2) F1: gearwheels are missing teeth and the outer ring of the left-side bearing is worn; (3) F2: gearwheels are missing teeth and inner ring of the left-side bearing is worn; (4) F3: gearwheels are missing teeth and the balls on the left-side bearing are missing; (5) F4: pinion and gearwheels are missing teeth; and (6) F5: object is in a normal state. The detailed information about the dataset we used in our experiments are provided in Table 5.

Table 5. Description of petrochemical dataset.

Fault Location		F0	F1	F2	F3	F4	
Fault Label		0	1	2	3	4	5
Dataset	Pretrain	500	500	500	500	500	500
	Unlabeled	1000	1000	1000	1000	1000	1000
	Test	200	200	200	200	200	200

4.2.2. Six-Way Fault Classification

The petrochemical dataset has six fault classes, and each sample has 1024 sampling points. In this case, study experiment, we chose 500 samples for training, and 1000 unlabeled samples and 200 labeled samples for evaluation. Detailed dataset information is listed in Table 5. The parameters of the model were the same as used for CWRU dataset. We determined the classification accuracy of all considered methods, as provided in Table 6.

Table 6. Classification accuracy of all considered methods.

Method	Petrochemical Dataset				
	1-Shot	5-Shot	10-Shot	15-Shot	30-Shot
CNN	41.46 ± 0.16	76.63 ± 0.41	92.84 ± 0.01	82.79 ± 0.14	91.78 ± 0.24
ProtoNet [24]	86.79 ± 0.53	94.07 ± 0.18	97.04 ± 0.06	97.56 ± 0.14	97.71 ± 0.29
IPN [28]	88.86 ± 2.20	89.12 ± 1.91	100	100	100
ProtoNet+KPCA	88.70 ± 0.38	95.35 ± 0.07	98.38 ± 0.07	98.30 ± 0.04	99.17 ± 0.20
PSSPN	89.61 ± 0.30	96.23 ± 0.06	97.77 ± 0.08	98.57 ± 0.04	97.30 ± 0.03

As shown in Table 6, compared with the other few-shot learning strategies, the lowest classification accuracy was obtained by CNN. However, both ProtoNet and IPN achieved accurate classification. In 10- and 30-shot, IPN reached 100% classification accuracy. Adding KPCA, the accuracy of slightly ProtoNet improved. For of one- and five-shot learning, PSSPN performed the best, showing that the model can deal with the situations when data are scarce.

The confusion matrix of five-shot classification accuracy is shown in Figure 8. Many samples were correctly classified, and the average classification accuracy was 96.2%.

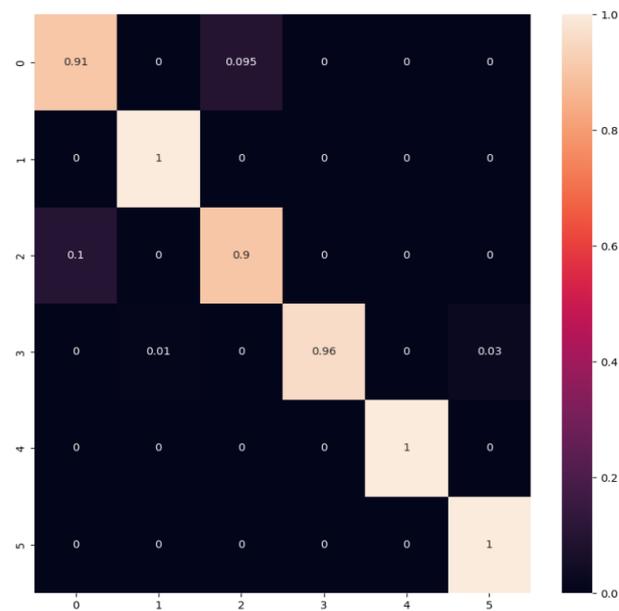


Figure 8. Confusion matrix results for 5-shot classification.

Figure 9 shows feature visualization via t-SNE, showing that CNN could not clearly distinguish different fault classes. ProtoNet and IPN performed better than CNN: the distance between features of different classes was as large as possible and the features in the same classes were close to each other. However, some features from different classes were still close to each other. Our proposed method is relatively more accurate than the others in t-SNE. In our method, different classes have well-defined boundaries.

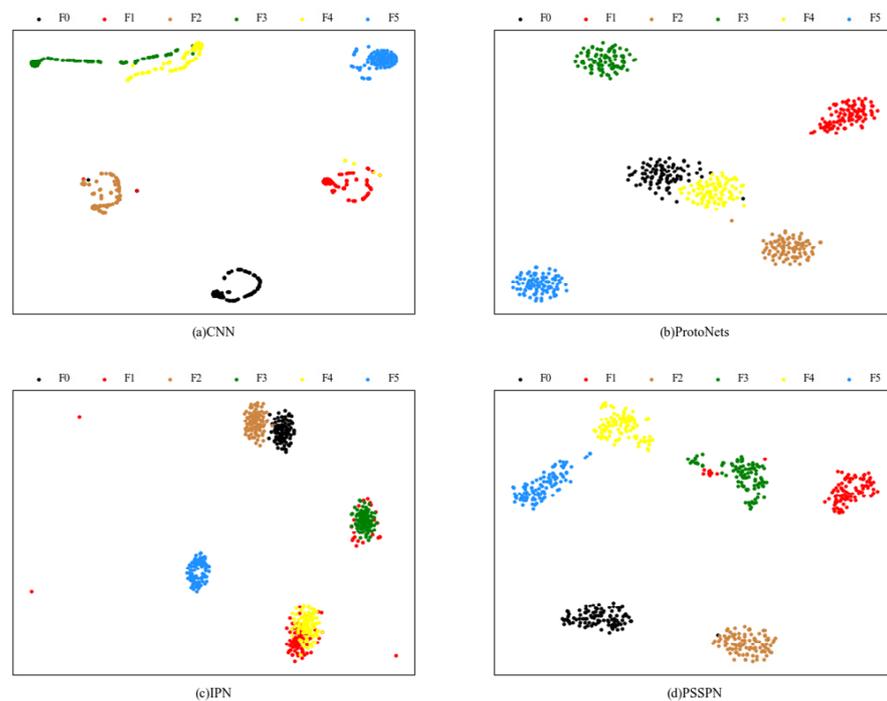


Figure 9. T-SNE visualization of different methods.

5. Conclusions

We presented a kernel principal component analysis method with a semi-supervised prototype network (PSSPN) for few-shot bearing fault diagnosis. This method can be used when few labeled samples are available and makes full use of the unlabeled data to train the

model. KPCA is used for avoiding the dimensionality problem to improve the accuracy of the classification results. We used pseudo-labeled data to fine-tune the pretrained model to avoid the problem of model overfitting. We used two datasets to evaluate the performance of the proposed method, and the results showed that compared with two other methods, the classification accuracy of our proposed method is higher when few labeled samples are available. In the future, we will improve the model to deal with data that are difficult to distinguish and increase the accuracy of the classification result.

Author Contributions: Methodology and Writing—Original draft preparation, J.H.; software and validation, Z.Z.; funding acquisition and supervision, X.F.; formal analysis, Y.C.; investigation and writing—review and editing, D.C.; resources and data curation, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China with grants 61703104 and 62173349; Key Laboratory of High Performance Complex Manufacturing with grant ZZYJKT2020-14; and Hunan Provincial Key Laboratory with grant 2017TP1002.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: MDPI Research Data Policies.

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

References

1. Liu, R.; Yang, B.; Zio, E.; Chen, X. Artificial intelligence for fault diagnosis of rotating machinery: A review. *Mech. Syst. Signal Process.* **2018**, *108*, 33–47. [[CrossRef](#)]
2. Zhou, D.H.; Zhao, Y.H.; Wang, Z.D.; He, X.; Gao, M. Review on Diagnosis Techniques for Intermittent Faults in Dynamic Systems. *IEEE Trans. Ind. Electron.* **2020**, *67*, 2337–2347. [[CrossRef](#)]
3. Rai, A.; Upadhyay, S.H. A review on signal processing techniques utilized in the fault diagnosis of rolling element bearings. *Tribol. Int.* **2016**, *96*, 289–306. [[CrossRef](#)]
4. He, J.; Ouyang, M.; Chen, Z.; Chen, D.; Liu, S. A Deep Transfer Learning Fault Diagnosis Method Based on WGAN and Minimum Singular Value for Non-Homologous Bearing. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–9. [[CrossRef](#)]
5. Qin, S.; Jiang, T. Improved Wasserstein conditional generative adversarial network speech enhancement. *EURASIP J. Wirel. Commun. Netw.* **2018**, *2018*, 181. [[CrossRef](#)]
6. Zhu, L.; Chen, Y.; Ghamisi, P.; Benediktsson, J.A. Generative Adversarial Networks for Hyperspectral Image Classification. *IEEE Trans. Geosci. Remote Sens.* **2018**, *56*, 5046–5063. [[CrossRef](#)]
7. Yin, H.; Li, Z.; Zuo, J.; Liu, H.; Yang, K.; Li, F. Wasserstein Generative Adversarial Network and Convolutional Neural Network (WG-CNN) for Bearing Fault Diagnosis. *Math. Probl. Eng.* **2020**, *2020*, 2604191. [[CrossRef](#)]
8. Gong, W.; Chen, H.; Zhang, Z.; Zhang, M.; Wang, R.; Guan, C.; Wang, Q. A novel deep learning method for intelligent fault diagnosis of rotating machinery based on improved CNN-SVM and multichannel data fusion. *Sensors* **2019**, *19*, 1693. [[CrossRef](#)]
9. Jiang, H.; Li, X.; Shao, H.; Zhao, K. Intelligent fault diagnosis of rolling bearings using an improved deep recurrent neural network. *Meas. Sci. Technol.* **2018**, *29*, 065107. [[CrossRef](#)]
10. Cui, M.; Wang, Y.; Lin, X.; Zhong, M. Fault diagnosis of rolling bearings based on an improved stack autoencoder and support vector machine. *IEEE Sens. J.* **2020**, *21*, 4927–4937. [[CrossRef](#)]
11. Zhang, C.; Zhang, Y.; Hu, C.; Liu, Z.; Cheng, L.; Zhou, Y. A novel intelligent fault diagnosis method based on variational mode decomposition and ensemble deep belief network. *IEEE Access* **2020**, *8*, 36293–36312. [[CrossRef](#)]
12. Zhang, A.; Li, S.; Cui, Y.; Yang, W.; Dong, R.; Hu, J. Limited data rolling bearing fault diagnosis with few-shot learning. *IEEE Access* **2019**, *7*, 110895–110904. [[CrossRef](#)]
13. Jiang, C.; Chen, H.; Xu, Q.; Wang, X. Few-shot fault diagnosis of rotating machinery with two-branch prototypical networks. *J. Intell. Manuf.* **2022**. [[CrossRef](#)]
14. Xu, J.; Xu, P.; Wei, Z.; Ding, X.; Shi, L. DC-NNMN: Across Components Fault Diagnosis Based on Deep Few-Shot Learning. *Shock Vib.* **2020**, *2020*, 3152174. [[CrossRef](#)]
15. Wang, D.; Zhang, M.; Xu, Y.; Lu, W.; Yang, J.; Zhang, T. Metric-based meta-learning model for few-shot fault diagnosis under multiple limited data conditions. *Mech. Syst. Signal Process.* **2021**, *155*, 107510. [[CrossRef](#)]
16. Xu, Y.; Li, Y.; Wang, Y.; Zhong, D.; Zhang, G. Improved few-shot learning method for transformer fault diagnosis based on approximation space and belief functions. *Expert Syst. Appl.* **2021**, *167*, 114105. [[CrossRef](#)]
17. Tao, X.; Ren, C.; Li, Q.; Guo, W.; Liu, R.; He, Q.; Zou, J. Bearing defect diagnosis based on semi-supervised kernel Local Fisher Discriminant Analysis using pseudo labels. *ISA Trans.* **2021**, *110*, 394–412. [[CrossRef](#)]

18. Zhang, X.; Su, Z.; Hu, X.; Han, Y.; Wang, S. Semi-supervised momentum prototype network for gearbox fault diagnosis under limited labeled samples. *IEEE Trans. Ind. Inform.* **2022**, *18*, 6203–6213. [[CrossRef](#)]
19. 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.* **2021**, *120*, 383–401. [[CrossRef](#)]
20. Huang, K.; Geng, J.; Jiang, W.; Deng, X.; Xu, Z. Pseudo-loss Confidence Metric for Semi-supervised Few-shot Learning. In Proceedings of the 18th IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 8651–8660.
21. Wang, D.; Han, S.; Wang, Q.; He, L.; Tian, Y.; Gao, X. Pseudo-Label Guided Collective Matrix Factorization for Multiview Clustering. *IEEE Trans. Cybern.* **2021**. [[CrossRef](#)]
22. Sung, F.; Yang, Y.; Zhang, L.; Xiang, T.; Torr, P.H.S.; Hospedales, T.M. Learning to Compare: Relation Network for Few-Shot Learning. In Proceedings of the 31st IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 1199–1208.
23. Pan, Y.; Yao, T.; Li, Y.; Wang, Y.; Ngo, C.-W.; Mei, T.; Soc, I.C. Transferrable Prototypical Networks for Unsupervised Domain Adaptation. In Proceedings of the 32nd IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 2234–2242.
24. Snell, J.; Swersky, K.; Zemel, R. Prototypical Networks for Few-shot Learning. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017.
25. Varon, C.; Alzate, C.; Suykens, J.A.K. Noise Level Estimation for Model Selection in Kernel PCA Denoising. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *26*, 2650–2663. [[CrossRef](#)] [[PubMed](#)]
26. Ji, Z.; Chai, X.; Yu, Y.; Pang, Y.; Zhang, Z. Improved prototypical networks for few-Shot learning. *Pattern Recognit. Lett.* **2020**, *140*, 81–87. [[CrossRef](#)]
27. Smith, W.A.; Randall, R.B. Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study. *Mech. Syst. Signal Process.* **2015**, *64–65*, 100–131. [[CrossRef](#)]
28. Tian, X.; Chen, L.; Zhang, X.; Chen, E. Improved prototypical network model for forest species classification in complex stand. *Remote Sens.* **2020**, *12*, 3839. [[CrossRef](#)]
29. Zhang, J.; Zhang, Q.; He, X.; Sun, G.; Zhou, D. Compound-Fault Diagnosis of Rotating Machinery: A Fused Imbalance Learning Method. *IEEE Trans. Control Syst. Technol.* **2021**, *29*, 1462–1474. [[CrossRef](#)]
30. Hu, Q.; Qin, A.; Zhang, Q.; He, J.; Sun, G. Fault Diagnosis Based on Weighted Extreme Learning Machine with Wavelet Packet Decomposition and KPCA. *IEEE Sens. J.* **2018**, *18*, 8472–8483. [[CrossRef](#)]