

Relation Network Using Metalearning for Intelligent Machinery Fault Diagnosis with Few Labeled Samples

Lin Fang,^{1*} Chuan Li,² Xin Wang,¹ and Cheng-Fu Yang^{3,4**}

¹School of Urban Construction and Intelligent Manufacturing, Dongguan City University, Dongguan 52300, China

²School of Mechanical Engineering, Dongguan University of Technology, Dongguan 523808, China

³Department of Chemical and Materials Engineering, National University of Kaohsiung, Kaohsiung 811, Taiwan

⁴Department of Aeronautical Engineering, Chaoyang University of Technology, Taichung 413, Taiwan

(Received January 3, 2024; accepted May 8, 2024)

Keywords: few-shot learning, relation network, end to end, high accuracy

In this paper, we introduce a novel technology utilizing relation networks with metalearning for intelligent machinery fault diagnosis, particularly in scenarios with limited labeled samples, employing the principles of few-shot learning (FSL). The proposed approach was characterized by its flexibility, simplicity, and a versatile framework. FSL facilitated the recognition and classification of new classes, requiring only a small number of samples from each category. The core of this method was the adaptive end-to-end training of the relation network (RN) from scratch. During the metalearning stage, the RN learned a deep distance metric, enabling the comparison of limited fault samples within episodes, and these episodes were simulated within the context of few-shot settings. Following the training process, the RN demonstrated the capability to classify samples from new classes by computing relation scores. Notably, it could also compare query samples with the limited samples from each new class without the need for further network updates. Experimental verification solidified the effectiveness of the proposed RN method, showcasing its robust classification abilities and achieving a relatively high level of accuracy. This technology holds promise for enhancing fault diagnosis in intelligent machinery, particularly in scenarios where labeled samples are scarce.

1. Introduction

Rotating machinery constitutes a cornerstone in various contemporary industrial sectors, as highlighted by Ma *et al.*⁽¹⁾ and Zhao and Lin.⁽²⁾ Bearings, serving as pivotal components within this machinery, wield considerable influence over the operational efficiency of mechanical equipment, as underscored by Shao *et al.*⁽³⁾ The ramifications of bearing failure are far-reaching, encompassing mechanical damage, substantial economic losses, and potential threats to user safety. This issue becomes particularly critical in the context of modern industries, where intricate and high-precision structures are prevalent, including those found in high-speed train motors, aero-engines, and wind turbine generators. In light of these considerations, the accurate

*Corresponding author: e-mail: 494812652@qq.com

**Corresponding author: e-mail: cfyang@nuk.edu.tw

<https://doi.org/10.18494/SAM4891>

prediction and fault diagnosis of various bearing operations assume paramount significance in both real industrial scenarios and academic research. The need for precision is underscored by the complex nature of contemporary machinery, necessitating a comprehensive understanding of bearing behavior for enhanced operational reliability and safety across diverse applications.

Typically, fault diagnosis methods can be categorized into two main approaches: model-based and data-based methods. The model-based approach involves monitoring and identifying abnormal states of equipment by simulating its normal functioning. In the model-based approach, the focus lies in creating a virtual representation of the equipment's normal operational state. This is achieved through simulation and modeling techniques, allowing for the identification of deviations from the expected behavior, which indicate potential faults or anomalies. In contrast, the data-based fault diagnosis method establishes a mapping relationship between data and labels, where the model is trained to recognize and map this relationship to determine the fault type. The data-driven fault diagnosis method generally consists of four sequential steps: signal processing, feature extraction, feature dimension reduction, and fault pattern recognition.

1.1 Model-based fault diagnosis

Model-based fault diagnosis relies on the application of physically degraded principles or system identification techniques to achieve high-accuracy monitoring. In a groundbreaking work by Stengel,⁽⁴⁾ a fault-tolerant control system was introduced in 1991, and the applications of intelligent failure-tolerant control systems were explored. The observer plays a pivotal role in model-based fault diagnosis, facilitating fault detection, isolation, and identification. The signals from the observer promote the influence of disturbance signals, which encompass modeling errors, process disturbances, and measurement noise, as part of the solution to this complex problem. In a related study, Patton *et al.* employed the parametric eigenstructure assignment strategy to detect sensor faults in aircraft.⁽⁵⁾ Additionally, Zhang *et al.* proposed a linear matrix inequality-based fault detection fuzzy observer applicable to various nonlinear systems.⁽⁶⁾ These examples underscore the versatility and effectiveness of model-based fault diagnosis methods in addressing complex monitoring challenges.

Stochastic approaches for fault diagnosis emerged in the 1970s as an alternative avenue of exploration. In a notable contribution, Li and Olson proposed fault detection and diagnosis in nonlinear systems through the utilization of extended Kalman filters.⁽⁷⁾ Particularly in dynamic systems where signals transition between distinct values, a phenomenon common in discrete event systems, stochastic methods became instrumental. The fault diagnosis of discrete event systems was introduced in the 1990s. Advancements in this domain continued with Dotoli *et al.* enhancing online fault detection in discrete event systems by employing Petri nets and integer linear programming.⁽⁸⁾ As technology rapidly progressed, the field of fault diagnosis expanded to encompass networked and distributed systems. However, challenges arose owing to limitations in sensor capacity, especially in wireless setups, leading to issues such as random data dropout, communication delays, and scheduling confusion. These challenges underscore the need for innovative solutions to ensure the reliability and effectiveness of fault diagnosis in modern, interconnected systems.

In a study conducted by He *et al.*,⁽⁹⁾ a direct state estimation approach, which integrates Kalman and least-square filters, was employed for the fault detection and diagnosis of networked sensing systems. This innovative method was aimed at enhancing the robustness and accuracy of fault identification in complex networked environments. Similarly, El-Zonkoly implemented distributed fault diagnosis specifically tailored for power networks, showcasing the adaptability of the fault diagnosis techniques across diverse industrial applications.⁽¹⁰⁾ While model-based fault diagnosis methods have achieved considerable success, their application faces challenges, particularly in deciphering fault mechanisms in complex equipment. The inherent difficulty in modeling intricate systems limits the universal applicability of the model-based approach. Consequently, with advancements in data processing and monitoring methods, organizations and researchers have increasingly turned to data-based fault diagnosis methods in industrial fields. This shift underscores the recognition of the practical advantages offered by data-driven approaches in tackling the complexities of real-world fault diagnosis scenarios.

1.2 Data-based fault diagnosis

Owing to the advancement of neural networks, many shallow machine learning techniques utilizing Fourier transformation to extract signals for acquiring frequency-domain and time-domain characteristics have experienced significant success. Shallow machine learning methods such as support vector machine, K -nearest neighbor algorithm, and random forest (RF) have captured the interest of researchers and organizations alike. Widodo and Yang applied these shallow machine learning methods to machine condition monitoring and fault diagnosis, demonstrating outstanding performance in generalization and high accuracy in classification.⁽¹¹⁾ The improved K -nearest neighbor approach was employed for intelligent fault diagnosis, employing sparse filtering to extract features and devising a case-based reconstruction algorithm for dynamically obtaining sample vectors.⁽¹²⁾ In another study, Cerrada *et al.* utilized genetic algorithms and an RF-based classifier to establish a robust system for multiclass fault diagnosis.⁽¹³⁾ However, these methods require modelers to design or manually select features, as machine learning lacks end-to-end learning. Similarly, adaptive learning of signal extraction features is not possible, necessitating considerable expertise to construct these features.

With the advancement of deep learning, numerous researchers have incorporated deep learning techniques into fault diagnosis. Deep learning encompasses end-to-end learning, allowing the system to derive typical features directly from raw data. This capability leads to the establishment of deep neural networks (DNNs) with intricate architectures. Deep learning excels in adaptively extracting features. Currently, deep learning methods, such as convolutional neural networks, the deep Boltzmann machine, recurrent neural networks (RNN), and autoencoders (AEs), are extensively employed in intelligent fault diagnosis. Shao *et al.* introduced an innovative approach for intelligent fault diagnosis in rolling bearings using deep ensemble AEs.⁽¹⁴⁾ This method eliminates the need for manual feature extraction, resulting in enhanced efficacy. Following this, they applied an enhanced deep feature fusion method to bolster feature learning capabilities through deep AEs, yielding positive results.⁽¹⁵⁾ Jia *et al.* utilized datasets from rolling element bearings and planetary gearboxes, as DNNs can adaptively extract relevant fault characteristics from signals.⁽¹⁶⁾

Despite its notable achievements, deep learning fault diagnosis has inherent limitations. It demands a substantial amount of labeled data and samples for training in each fault mode to establish fault classifications. While deep learning fault modes can be trained using abundant sample data from a single fault class in a controlled laboratory environment, practical applications often lead to the emergence of new faults due to changes in working conditions. These new faults are typically not accounted for in laboratory experiment simulations. Consequently, models need to be updated, and datasets must be retrained to identify these new fault categories. In reality, the time constraints of collecting ample samples for new faults are a challenge, often resulting in only a limited amount of available fault sample data. Humans can leverage prior knowledge to identify a new object using only one or a few instances. Inspired by this ability, few-shot learning (FSL) has been introduced to learn data features in scenarios with limited examples, utilizing existing knowledge to address prediction and classification challenges in situations with minimal data. FSL can be categorized into data, models, and algorithms. Ren *et al.* introduced a capsule AE model, known as CaAE, based on an AE and capsule network for intelligent fault diagnosis.⁽¹⁷⁾ This model reduces the CaAE's dependence on samples by extracting feature capsules. Hu *et al.* proposed a self-adaptive convolutional neural network for fault diagnosis, incorporating a data augmentation algorithm based on the core assumption of order tracking.⁽¹⁸⁾ This network exhibits strong adaptability for FSL scenarios.

Xu *et al.* employed a deep convolutional nearest neighbor matching network (DC-NNMN) based on FSL.⁽¹⁹⁾ The DC-NNMN achieved fault diagnosis accuracies of 82.63% for bearings and 82.19% for gearing with just one sample from each fault category. The results illustrated the advantages of metalearning in few-shot fault diagnosis instances. Metalearning has emerged as a prominent research area in the field of artificial intelligence (AI). The primary goal is to develop a versatile AI model capable of learning various tasks from scratch without the need for extensive training. Currently, many researchers are exploring the applications of metalearning in FSL scenarios. Numerous fault diagnosis approaches adopt a metalearning or learning-to-learn strategy to extract transferable knowledge from a set of secondary tasks. This approach aids in effectively addressing the challenges of few-shot fault diagnosis.

Nevertheless, the conventional FSL methods often require intricate inference mechanisms, complex RNN architectures, or fine-tuning of the target, which can pose challenges in terms of complexity and efficiency. In this context, a transferrable deep metric learns and compares relationships between fault diagnoses. To enhance the learning process and promote generalizability, in this paper, we introduce a two-branch relation network (RN) designed for few-shot recognition. This network learns to contrast query fault diagnoses against few-shot labeled sample fault diagnoses. The methodology involves the generation of embedding modules for both query and training samples, followed by a contrastive analysis using a relation module to determine if they belong to similar categories. The FSL process is executed through end-to-end metalearning, facilitated by the embedding and relation modules. A learnable nonlinear comparator, viewed as an extension of the strategy, is employed. This proposed method outperforms previous approaches and offers simplicity and speed without the need for RNNs or fine-tuning. Furthermore, the proposed fault diagnosis method is more convenient and effective.

In comparison with existing methods, the approach presented in this study demonstrates innovation in three key aspects.

1. Many deep-learning-based fault diagnosis models currently require a substantial amount of labeled data, rendering them impractical for constructing new models with limited labeled data. The fault diagnosis model proposed in this paper, based on the FSL paradigm, effectively resolves this challenge.
2. In consideration of the characteristics of fault diagnosis problems, the RN method is employed to construct a fault diagnosis model with few samples, offering a tailored solution to the specific challenges posed by limited data.
3. The few-shot fault diagnosis method based on RN, as constructed in this study, has demonstrated outstanding diagnostic effectiveness across multiple datasets.
4. The remainder of this paper is structured as follows. A detailed description of the proposed method is provided in Sect. 2. In Sect. 3, bearing data is utilized to validate the effectiveness of the proposed approach. Finally, in Sect. 4, we conclude the paper and outline potential future directions.

2. Methodology

2.1 Application scenarios

In the past, conventional deep learning methods focused on directly acquiring knowledge about new fault categories, requiring a substantial amount of data in practical applications. However, in online fault diagnosis, new fault categories and samples emerge, even when these new fault samples are limited or singular. The primary challenge lies in training a fault diagnosis model effectively with a reduced number of samples. To address this issue, our laboratory possesses extensive datasets, with each class representing a distinct fault category. We aim to develop a method to leverage the knowledge gained from previous studies on diverse datasets to enhance the overall model's ability to identify new fault categories. For instance, suppose there were initially 10 fault categories and samples in the laboratory. In the subsequent phase, the 11th and 12th categories emerge, each with 10 samples. Discriminating these new fault types is referred to as FSL. The FSL approach involves creating a model that trains on the existing 10 fault categories to recognize samples from the new 11th and 12th categories, utilizing knowledge transfer from samples of previous classes to facilitate learning about the new categories.

To address this challenge, the dataset is typically partitioned into three categories for implementing few-shot fault diagnosis: the training set, support set, and testing set. The training set is a distinct set for FSL in conjunction with the testing set. The support set comprises labeled samples from the new categories, whereas the testing set requires classification. Fault samples from the support set are termed support fault samples, which are analogous to the query fault samples as defined. Standardization is articulated in terms of “way” and “shot”. “Way” denotes the number of new categories to be identified in a single test process, whereas “shot” represents the number of support fault samples for each category. We opted for an equal number of support fault samples for each category, resulting in the utilization of the N -way K -shot paradigm in FSL

scenarios. For instance, in a scenario with two new categories, each containing ten samples, three samples from each category were chosen to form the support set. Subsequently, three of the remaining seven samples were selected to constitute the testing set, distinct from the support set. This type of problem is commonly referred to as a “two-way three-shot” problem.

Figure 1 illustrates the N -way K -shot classification problem. Specifically, Fig. 1(a) addresses FSL, whereas Figs. 1(b) and 1(c) represent the original learning training stages. FSL is a form of metalearning with the goal of achieving “learn to learn”, enabling the computer to autonomously acquire learning capabilities. In Fig. 1(a), the source domain’s training set contains a sufficient number of samples to train a neural network. However, the support set in the target domain is a small sample set, insufficient for training a large neural network. Initially training on fault categories in the source domain’s training set allows the model to discern similarities and differences between the source and target domains. This enables the model to identify fault categories in the target domain, selecting some faults as support sets and testing sets. Transitioning to Fig. 1(b), during the metalearning training phase, a subset of x samples from X categories in the training set is chosen as the sample set. Simultaneously, another set of x samples from different samples is selected as the query set for training the model. In Fig. 1(c), during the

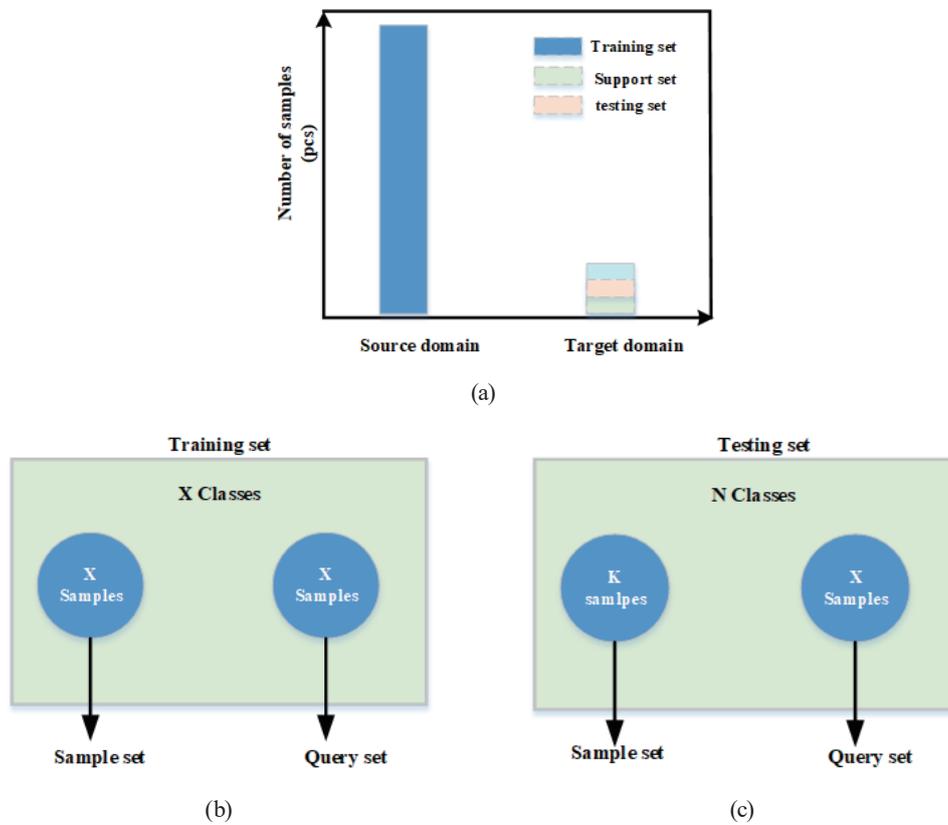


Fig. 1. (Color online) N -way K -shot classification problem. (a) FSL, (b) metatraining stage, and (c) metatesting stage.

metalearning testing phase, x samples are chosen from N categories as the sample set, and another set of x samples from different samples is selected as the query set.

2.2 Problem definition

A classifier can be trained to assign a class label \hat{y} to each sample \hat{x} in the test set, utilizing the support set theoretically. However, owing to the limited labeled samples in the support set, the classifier's performance tends to be suboptimal. To address this issue, we focused on implementing metalearning on the training set. This approach is aimed at extracting transferrable knowledge that enhances FSL on the support set, resulting in the improved classification of the test set. To conduct FSL training on the training set, the authors adopted the effective strategy of episode-based training. In each training iteration, an episode involves stochastically selecting N classes from the training set, with K labeled samples from each of the N classes forming the sample set $D = \{(x_i, y_i)\}_{i=1}^m$ ($m = K \times N$). The remainder of those N classes' samples acts as the query set $Q = \{(x_i, y_i)\}_{i=1}^n$. The sample/query set is deliberately separated into a simulated support set and a test set, mirroring the conditions encountered during testing. When required, a model trained on a sample/query set can undergo fine-tuning using the support set. To address this challenge, the authors employed an episode-based training strategy. In the few-shot experiments outlined in Sect. 3.1, both one-shot ($K = 1$) and five-shot ($K = 5$) settings were utilized, as depicted in Fig. 2.

The RN in this study consists of an embedding module f_θ and a relation module g_θ , as illustrated in Fig. 3. The embedding module f_θ is applied to the samples x_b in query set Q and the samples x_a in sample set D . This results in the generation of the feature maps $f_\theta(x_b)$ and $f_\theta(x_a)$ for the query set and sample set, respectively. The feature maps $f_\theta(x_a)$ and $f_\theta(x_b)$ are combined using the operator $N(f_\theta(x_a), f_\theta(x_b))$, with the feature maps assumed to be connected through the depth via $N(\cdot, \cdot)$. Subsequently, the relation module g_θ is fed with the composed feature map of the query and sample, producing a scalar in the range of 0–1. This scalar represents the relation score, ultimately indicating the similarity between x_a and x_b . In the N -way one-shot setting, N relation scores $t_{a,b}$ are generated, representing the relationship between one query input x_b and examples x_a from the training sample set:

$$t_{a,b} = g_\theta \left(N \left(f_\theta(x_a), f_\theta(x_b) \right) \right), a = 1, 2, \dots, N. \quad (1)$$

In the K -shot scenario (when $K > 1$), the feature map for a particular class is formed through an elementwise sum of the outputs from the embedding module for each sample within that training class. This aggregated class-level feature map is then combined with the feature map of the query sample, following the previously described process. Consequently, regardless of the few-shot or one-shot settings, the number of relationship scores for a single query is always N . For the objective function, the mean square error (MSE) loss, as defined in Eq. (2), was employed to train the model and regress the relation score $t_{a,b}$ to the ground truth. In this context, matching pairs have a similarity of 1, whereas mismatches are assigned a similarity of 0.

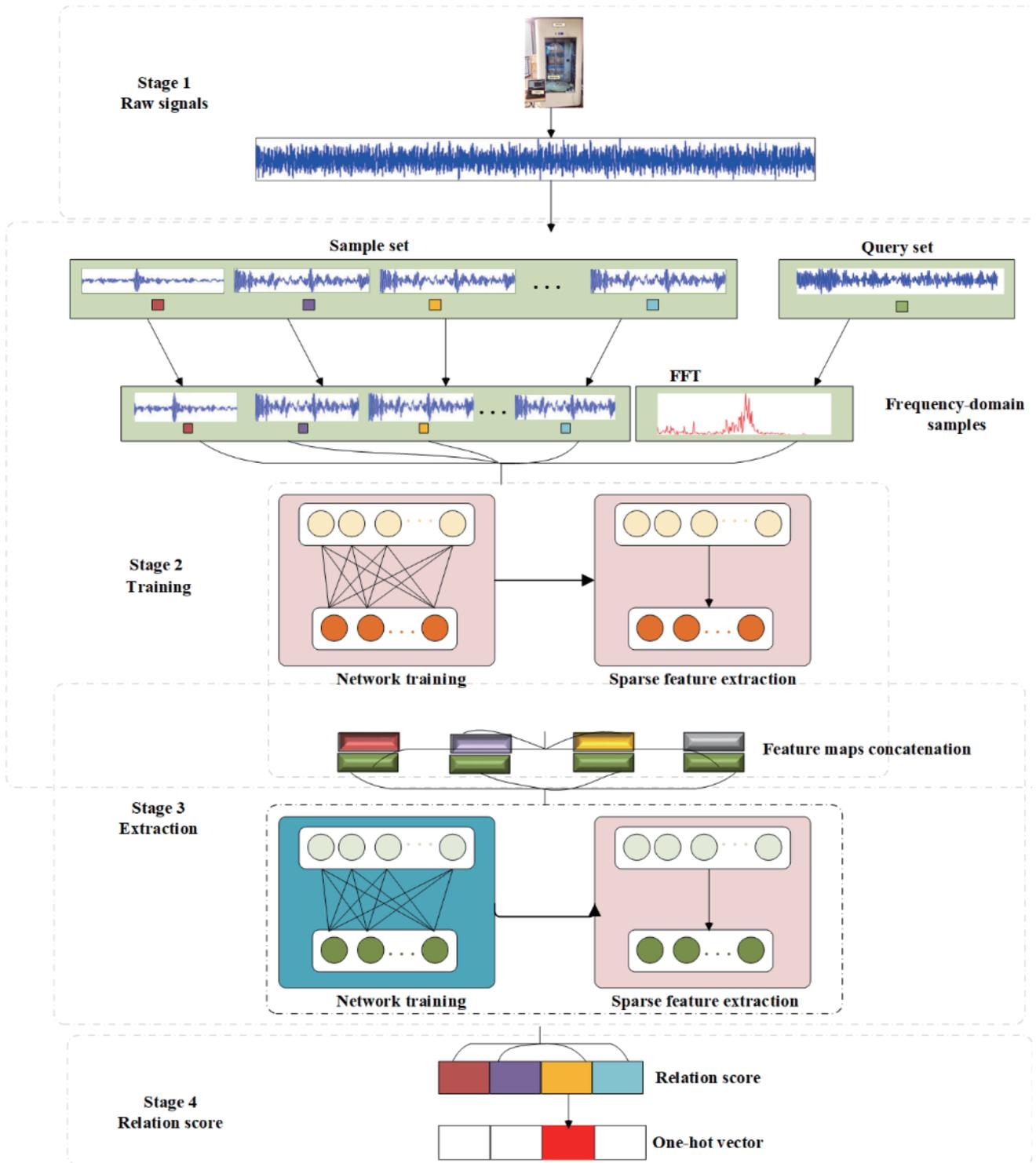


Fig. 2. (Color online) RN architecture for a five-way one-shot problem with a convolutional network.

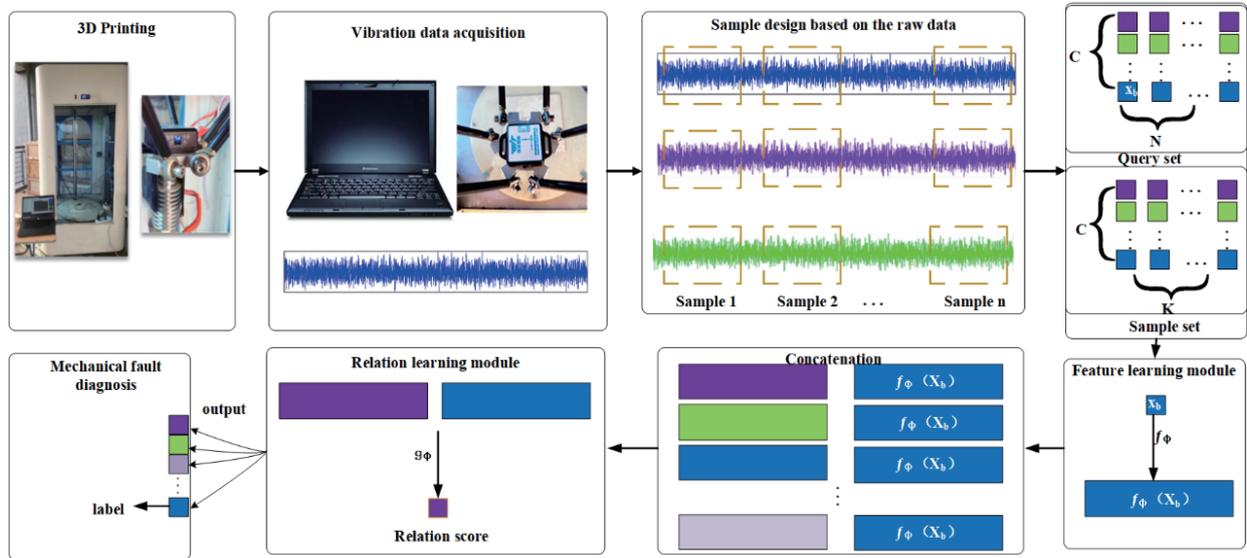


Fig. 3. (Color online) Flowchart of the fault diagnosis procedure.

$$\varphi, \Theta \leftarrow \underset{\varphi, \Theta}{\operatorname{argmin}} \sum_{a=1}^m \sum_{b=1}^n (t_{a,b} - 1(y_a == y_b))^2 \quad (2)$$

The use of MSE as the loss function is somewhat unconventional. The issue arises from the fact that the label space is binary, with values $\{0,1\}$, making the application of MSE less standard for classification tasks. Although relationship scores, by definition, are treated as predictions in a regression context, the assignment of $\{0,1\}$ targets for ground truth can be done arbitrarily.

3. Experiments

The methodology in this study was assessed across two interconnected tasks: few-shot classification using Case Western Reserve University (CWRU) bearing data and classification within the realm of 3D printing.⁽²⁰⁾ All experiments were conducted using the PyTorch framework. The fault diagnosis procedure is illustrated in Fig. 3. The process of collecting vibration data from mechanical equipment involves extracting samples from raw information. A feature learning model is established within both the sample set and query set. By training models in both the source and target domains, differences and similarities are discerned, resulting in the creation of a relationship learning model. Ultimately, this model generates scores for various types of mechanical fault, aiding in the identification of specific failure categories.

3.1 CWRU

3.1.1 Data description

The bearing data utilized in this study were provided by CWRU and were collected from a motor-driven mechanical system under four different loads, with sampling frequencies of 12 and 48 kHz. The bearing dataset encompasses conditions classified as follows: (1) normal conditions (N), (2) inner race fault (IF), (3) outer race fault (OF), and (4) ball faults (RF). These faults occur in the drive end bearing of the motor, with fault diameters of 0.18, 0.36, and 0.54 mm, respectively. Four bearing datasets, labeled A–D, were employed to evaluate the proposed method's performance. Table 1 provides a detailed description of the datasets. Datasets A, B, C, and D correspond to bearing categories with load ratings of 0, 1, 2, and 3 hp, respectively. The outer race positions relative to the load zone (centered at 6:00) in the @3, @6, and @12 o'clock positions have bearing diameters of 0.7, 1.4, and 2.1 in., respectively. There are four types of normal condition and 116 types of fault. In total, 76 categories were designated as the training set, each comprising 200 samples. The support dataset comprises 12 categories with 20 samples, and the testing dataset includes 28 categories with 20 samples.

3.1.2 CWRU dataset

The CWRU dataset consists of 16,000 samples distributed across 116 classes. The data split used in this study follows the division introduced in a prior investigation, with 76, 12, and 28 classes allocated for training, validation, and testing, respectively.⁽²¹⁾ Each training set comprises 200 samples, and each validation and test set includes 20 samples per class. The 28 validation classes are specifically utilized for monitoring the generalization performance.

Table 1
Description of bearing datasets.

Fault diameter (In)	Motor load (HP)	Inner race	Ball	Outer race		
				Position relative to the load zone (Load zone centered at 6:00)		
				Centered @6:00	Orthogonal @3:00	Opposite @12:00
0.007"	0	IR007_0	B007_0	OR007@6_0	OR007@3_0	OR007@12_0
	1	IR007_1	B007_1	OR007@6_1	OR007@3_1	OR007@12_1
	2	IR007_2	B007_2	OR007@6_2	OR007@3_2	OR007@12_2
	3	IR007_3	B007_3	OR007@6_3	OR007@3_3	OR007@12_3
0.014"	0	IR014_0	B014_0	OR014@6_0	*	*
	1	IR014_1	B014_1	OR014@6_1	*	*
	2	IR014_2	B014_2	OR014@6_2	*	*
	3	IR014_3	B014_3	OR014@6_3	*	*
0.021"	0	IR021_0	B021_0	OR021@6_0	OR021@3_0	OR021@12_0
	1	IR021_1	B021_1	OR021@6_1	OR021@3_1	OR021@12_1
	2	IR021_2	B021_2	OR021@6_2	OR021@3_2	OR021@12_2
	3	IR021_3	B021_3	OR021@6_3	OR021@3_3	OR021@12_3

3.1.3 Training

Following the standard protocol employed by most existing FSL studies, sampling was performed as follows. In addition to the K samples, five-way one-shot and five-way five-shot classifications were carried out. The five-way one-shot classification involved five queries, and the five-way five-shot classification had five queries for each of the N sampled classes in each validation episode. This setup resulted in 30 samples in one validation episode for the five-way one-shot example ($5 \times 5 + 1 \times 5$), and 50 samples in one validation episode for the five-way five-shot example ($5 \times 5 + 5 \times 5$). Similarly, the five-way one-shot classification included five queries, and the five-way five-shot classification comprised five queries for each of the N sampled classes in each testing episode. This configuration led to 30 samples in one testing episode for the five-way one-shot example and 50 samples in one testing episode for the five-way five-shot example. The training process was executed over 50 epochs, with a dataset encompassing 5000 episodes. The model in this study was trained end to end from scratch, initialized at a rate of 0.001. The stochastic gradient descent method utilized Adam, with the learning rate halved every 2000 episodes.

3.2 Network architecture

As most embedding modules in FSL models utilize four convolutional blocks, the identical settings depicted in Fig. 4(a) were employed for a fair architectural comparison. In the experiment, accelerated datasets were configured with three channels (x , y , and z) representing elemental vibration signals. The time series signal was segmented into time domain samples,

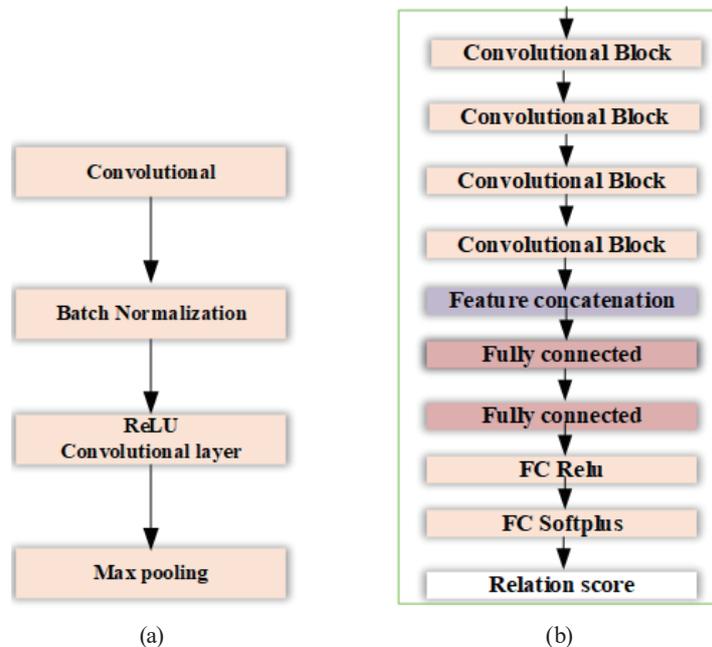


Fig. 4. (Color online) RN architecture for FSL: (a) convolutional block and (b) RN for FSL.

with each sample consisting of 512 sampling points, resulting in a sample size of 3×512 . The 1D convolution block included a 3×3 1D convolution layer, a batch layer, a rectified linear unit (ReLU) activation function, and a 2×2 maximum pooling layer. The convolution layer had a stride set to 1, padding set to 1, and the number of convolution kernels adjusted in accordance with the situation. Specifically, four 1D convolution blocks were tailored for the time domain feature extraction module. The first two convolution blocks had 16 dimensions for the convolution layer's number of kernels, whereas the last two blocks had eight dimensions. The variance estimation module comprised three convolution blocks, with the number of convolution kernels being 32 and 1, respectively.

In the architecture, a batch normalization layer, ReLU activation function, and a 2×2 maximum pooling layer were incorporated. The output size of the final maximum pooling layer, denoted as H, was set to 32, and for CWRU Net, H was adjusted to $32 \times 4 = 128$. The output layer exclusively utilized Softplus activation. All fully connected layers employed ReLU activation to generate relational scores within a reasonable range for our network architecture. Figure 4(b) illustrates the RN used in FSL. A total of 5000 sets of scenarios were trained in the CWRU experiment. The Adam optimization algorithm was employed for parameter optimization, with initial learning rates set at 0.15, 0.6, and 0.25. After every 100 training iterations, the model was validated on the verification set. The best model, confirmed through 50 consecutive validations, was selected as the final model for testing. Subsequent to training the embedding models, few-shot classification accuracies were computed on the CWRU dataset by averaging over 5,000 randomly generated episodes from the testing set. In both one-shot and five-shot scenario experiments, five query samples per class were batched in each episode for evaluation.

3.3 3D printer

In this section, a testbed for the 3D printer setup is established to conduct three-axis velocity and three-axis angle measurements. This testbed is employed to evaluate the effectiveness of the proposed RN technique for fault detection. The experiments are conducted to draw comparisons with the CWRU models.

3.3.1 Data description

The program's performance was assessed using experimental devices from 3D printers. The 3D printer under consideration, denoted as BWT901, features three degrees of freedom, operates with a synchronous belt drive, and adopts a delta kinematic configuration, as illustrated in Fig. 5. The printer incorporates a belt drive mechanism with a triangular motion configuration, where three stepper motors drive the belt for movement along the x , y , and z directions. An arm, formed by two metal rods, is equipped with bearings at its end, enabling rotational motion. The BWT901 attitude sensor, positioned at the base of the triangular configuration, captures angle signals, velocity signals, and magnetic field data for each axis. The sensor's sampling frequency is 100 Hz. The device is equipped with an analogue-to-digital conversion system, utilizing a USB



Fig. 5. (Color online) Test bed of the 3D printer. (a) Components of the experimental platform and (b) base of delta kinematic configuration with a coupled attitude sensor.

interface for collecting digital samples. Notably, the low-cost BWT901 attitude sensor may exhibit lower accuracy, resulting in noise within the collected data. However, the presence of noisy data proves beneficial for verifying and testing the performance of the fault diagnosis algorithm.

The tests were conducted during the trial stage, encompassing both pre-printing and post-printing periods, to validate the accuracy of the study's assessments and avoid interference during equipment operation. Fault detection before the test could impact print quality, and faults might occur during the printing process after the test. While additional test stages could have been explored, they were not within the scope of this study. During the test stage, the printer executed a cycling movement to generate data for the model. Cylindrical shell models with dimensions $R = 75$ mm and a height of 0.3 mm were printed. This operation was repeated 20 times, and the entire printing process was tested three times. Following the signal collection mentioned earlier, random printing times were uniformly distributed between 1 and 60 s. The computer captured the time series of 3D velocity and 3D angle signals, with each time series comprising 32400 samples.

3.3.2 Case study 1: joint bearings

The faults in 3D printers typically lead to a degradation in product quality. In this study, we focus on detecting connected bearing faults caused by the loosening of fastening screws, a type of fault that often arises from prolonged machine use. Thread loosening may manifest at either end of the arm or in any member connected to the arm. Accordingly, 12 fault scenarios ($3 \text{ arms} \times 2 \text{ ends} \times 2 \text{ rods}$ plus the normal state, designated as P) were designed. Each fault is induced by

the presence of loose screws. Experimental settings: The 3D printer is equipped with 12 joint bearings, each exhibiting a loosening of 0.25, 0.50, 0.75, 1.0, 1.25, 1.5, 1.75, and 2.0 turns. A total of 12 fault types were implemented in the experiment to simulate various working conditions, with different degrees of loosening faults. Figure 6 illustrates an example of a fault in a joint bearing, where Fig. 6(a) depicts a healthy joint bearing, and Fig. 6(b) shows an artificial faulty joint bearing with loosened fastening screws. In each fault mode, circles with a radius of R75 mm were printed, and data collection was repeated three times.

3.3.3 Case study 2: belts

The second case study involves the fault detection of the synchronous belt. This type of failure is also attributed to the long-time use of the machine. Owing to the elastic nature of the belt, it becomes increasingly longer, leading to severe deformation under stress. In addition, it will produce violent vibration under emergency braking, leading to the final product quality being low. The belt is loosened from its optimal clamping position to obtain the dataset under such faults. one, two, or three teeth are artificially loosened in each synchronous belt (Fig. 7). Three types of faults, denoted as M, N, and O, are respectively installed on arms 1, 2, and 3, while data without faults are labeled as P. Table 2 presents the fault modes, locations, and severity levels associated with each.

4. Simulation Results and Discussion

4.1 3D printer dataset

Table 2 shows the fault types of the 3D printer, where the bearing is categorized into 96 fault classifications based on fault location and degree. From these, a total of 62 classes were randomly

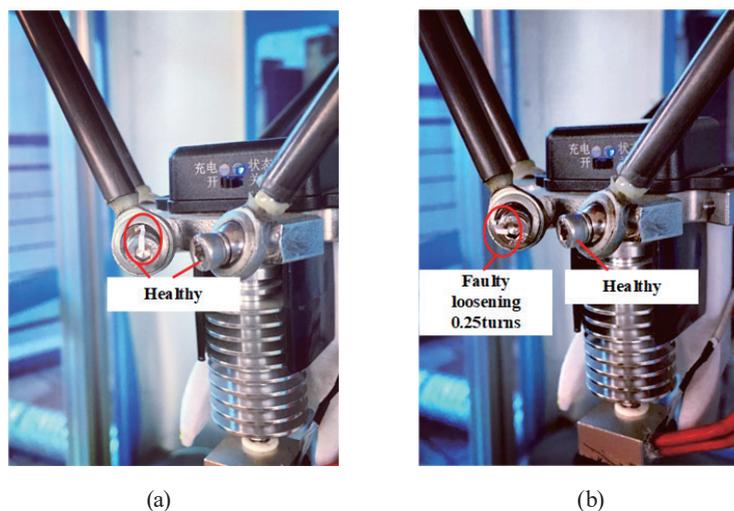


Fig. 6. (Color online) Example of fault in joint bearing. (a) Healthy joint bearing and (b) artificial faulty joint bearing (loosening of the fastening screws).

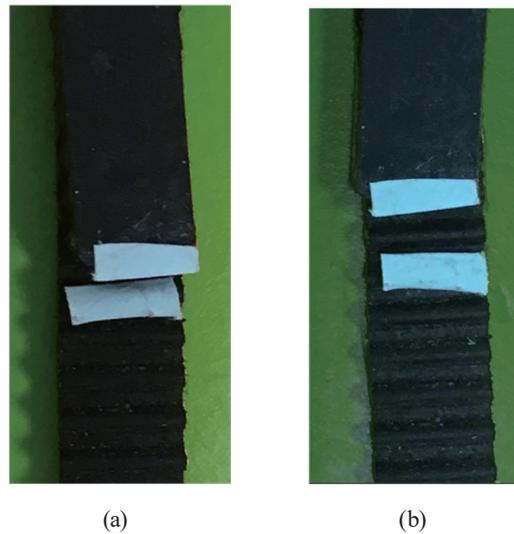


Fig. 7. (Color online) Example of fault in a synchronization belt. (a) Healthy belt and (b) artificial faulty belt (loosening one tooth).

Table 2
Fault type of the 3D printer.

Fault mode no.	Fault location	Fault degree
A	Joint bearings a	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
B	Joint bearings b	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
C	Joint bearings c	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
D	Joint bearings d	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
E	Joint bearings e	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
F	Joint bearings f	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
G	Joint bearings g	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
H	Joint bearings h	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
I	Joint bearings i	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
J	Joint bearings j	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
K	Joint bearings k	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
L	Joint bearings l	loosening0.25/0.5/0.75/1.0/1.25/1.5/1.75/2.0turns
M	Belt a	loosening 1/2/3 teeth
N	Belt b	loosening 1/2/3 teeth
O	Belt c	loosening 1/2/3 teeth
P	Normal	Normal

chosen to constitute the basic dataset (N_{base}), with each class containing 150 samples. The remaining classes were randomly allocated to form 10 verification sets and 24 testing sets, each comprising 20 samples. During the experiment, the acceleration data from the three channels (x , y , z) were utilized as the original vibration signal in the time domain. Specifically, 512 sampling points were extracted at the same position for each of the three channels, forming a time-domain sample with dimensions 512×3 . Subsequently, fast Fourier transformation was applied to each channel of the time-domain signal, resulting in corresponding samples in the frequency domain with dimensions 256×3 .

4.2 Difference between simulation and experimental results

The InfoPatch algorithm, recognized as an advanced small sample learning algorithm, was incorporated in this study for a comparative analysis to assess the performance of the proposed relational network. Figure 8 illustrates few-shot classification accuracies in five-way K -shot scenarios. Figure 8(a) portrays the metatask for training in the five-way one-shot setting, whereas Fig. 8(b) illustrates the metatask for training in the five-way five-shot scenario. Two variations of five-way K -shot learning tasks were devised for both CWRU and 3D printer datasets. The accuracies of the RN and InfoPatch demonstrated improvement with increasing K value. In the five-way five-shot experiments, the accuracies for the CWRU dataset were 90.7 ± 1.51 and 93.25 ± 0.63 , whereas for the 3D printer dataset, they were 91.02 ± 0.61 and 80.12 ± 0.14 , respectively [Fig. 8(a)]. In the five-way one-shot experiments, the accuracies for the CWRU dataset were 85.77 ± 0.68 and 86.85 ± 0.13 , and for the 3D printer dataset, they were 85.13 ± 0.21 and 70.14 ± 0.82 , respectively [Fig. 8(b)]. In both CWRU and 3D printer experiments, the few-shot classification accuracies are averaged over 100 test episodes. Table 3 provides few-shot classification accuracies on the CWRU dataset, and Table 4 shows few-shot classification accuracies on the 3D printer dataset.

Figure 9 depicts the cylindrical solid form of R75 generated by the 3D printer. In the solid structure, areas highlighted in red indicate locations where noticeable issues with print quality are observed. The looseness of the bearings and belt has a substantial impact on the formation of the solid object.

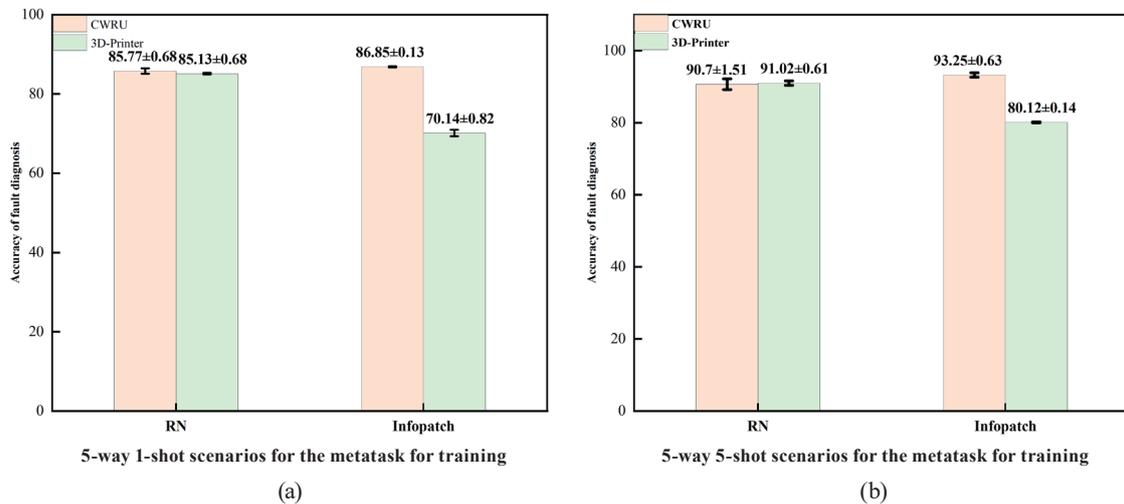


Fig. 8. (Color online) Few-shot classification accuracies on five-way K -shot scenarios. (a) Five-way one-shot scenarios for the metatask for training and (b) five-way five-shot scenarios for the metatask for training.

Table 3
Few-shot classification accuracies on CWRU dataset.

Model	Fine Tune	Five-way Accuracy (%)	
		Five-way one-shot	Five-way five-shot
RN	N	85.77 ± 0.68	90.7 ± 1.51
InfoPath	N	86.85 ± 0.13	93.25 ± 0.63

Table 4
Few-shot classification accuracies on the 3D printer dataset.

Model	Fine Tune	Five-way Accuracy (%)	
		Five-way one-shot	Five-way five-shot
RN	N	83.13 \pm 0.21	91.02 \pm 0.61
InfoPath	N	70.14 \pm 0.82	91.02 \pm 0.61



Fig. 9. (Color online) Resulting product generated by a faulty 3D printer.

5. Conclusions

In this study, we presented a straightforward “relationship network” method tailored for FSL scenarios. The essence of relational network learning lies in the comparison of embeddings between query and sample items utilizing a depth nonlinear distance metric. Through end-to-end training with contextual training, the network facilitates the fine-tuning of embeddings and distance metrics, enabling effective one-shot learning. The efficacy of the proposed method was experimentally examined using CWRU and 3D printer datasets, with a comparative analysis with the InfoPatch algorithm. In the CWRU dataset, the InfoPatch algorithm demonstrated marginally higher accuracy than the RN. However, in the 3D printer dataset, the RN algorithm exhibited a noteworthy 10% increase in accuracy over the InfoPatch average. This disparity underscores the effectiveness of the proposed method, particularly in scenarios with regular pulse settings. This result suggests that the “RN” method may exhibit superior performance in specific real-world applications, such as those encountered in 3D printing, where the RN’s adaptability and robustness lead to enhanced accuracy compared with existing algorithms such as InfoPatch. Further exploration and experimentation in diverse application domains could provide valuable insights into the versatility and generalizability of the proposed approach.

Acknowledgments

This research was supported by the education department of Fujian Province under Grant No. JAT210232, Summit-Tech Resource Corp., and projects under Grant Nos. MOST 111-2221-E-390-018 and NSTC 112-2622-E-390-002. This research was funded by GUANGDONG UNIVERSITY RESEARCH PROJECT, Grant No. 2022KTSCX219, and YUEJIAO GAO HAN, Grant No. [2021]4.

References

- 1 M. Ma, C. Sun, and X. Chen X: IEEE Trans. Industr. Inform. **14** (2018) 1137.
- 2 M. Zhao and J. Lin: Trans. Ind. Electron. **65** (2017) 2548.
- 3 H. Shao, H. Jiang, X. Li, and S. Wu: Knowl. Based Syst. **140** (2018) 1.
- 4 R. F. Stengel: Proc. 5th IEEE Int. Symp. Intelligent Control 1(Philadelphia, PA, USA, 1990) 548–557.
- 5 R. J. Patton, S. W. Willcox, and J. S. Winter: J. Guid. Control Dyn. **10** (1987) 359.
- 6 D. Zhang, H. Wang, B. Lu, and Z. Wang: Int. J. Innov. Comput. Inf. Control **8** (2012) 633.
- 7 R. Li R and J.H. Olson: Ind. Eng. Chem. Res. **30** (1991) 898.
- 8 M. Dotoli, M.P. Fanti, A. M. Mangini, and W. Ukovich: Automatica **45** (2009) 2665.
- 9 X. He, Z. Wang, Y. Liu, and D. H. Zhou: IEEE Trans. Industr. Inform. **9** (2013) 1670.
- 10 A. M. El-Zonkoly: Electr. Power Syst. Res. **81** (2011) 1482.
- 11 A. Widodo and B. S. Yang: Mech. Syst. Signal Process. **21** (2007) 2560.
- 12 J. Lu, W. Qian, S. Li, and R. Cui: Appl. Sci. **11** (2021) 919.
- 13 M. Cerrada, G. Zurita, D. Cabrera, R. V. Sánchez, M. Artés, and C. Li: Mech. Syst. Signal Process. **70** (2016) 87.
- 14 H. Shao, H. Jiang, Y. Lin, and X. Li: Mech. Syst. Signal Process. **102** (2018) 278.
- 15 H. Shao, H. Jiang, F. Wang, and H. Zhao: Knowl. Based Syst. **119** (2017) 200.
- 16 F. Jia, Y. Lei, J. Lin, X. Zhou, and N. Lu: Mech. Syst. Signal Process. **72** (2016) 303.
- 17 Z. Ren, Y. Zhu, K. Yan, K. Chen, W. Kang, Y. Yue, and D. Gao: Mech. Syst. Signal Process. **138** (2020) 106608.
- 18 T. Hu, T. Tang, R. Lin, M. Chen, S. Han, and J. Wu: Measurement. **156** (2020) 107539.
- 19 J. Xu, P. Xu, Z. Wei, X. Ding, and L. Shi: Shock Vib. **166** (2020) 108202.
- 20 Case School of Engineering, Case Western Reserve University: <https://engineering.case.edu/bearingdatacenter/download-data-file> (Accessed December 2022).
- 21 O. Vinyals, C. Blundell, T. Lillicrap, and D. Wierstra: Adv. Neural Inf. Process. Syst. **29** (2016) 1606.