

A Class-incremental Learning Method based on Knowledge Distillation for Multi-fault Diagnosis of Axial Piston Pump

Dandan Wang, Shihao Liu, Junhui Zhang*, Fei Lyu, Weidi Huang, and Bing Xu

State Key Laboratory of Fluid Power and Mechatronic System, Zhejiang University, Hangzhou, China
E-mail: ddwang@zju.edu.cn, shihaoliu@zju.edu.cn, benzjh@zju.edu.cn, feilv@zju.edu.cn, wduhuang@zju.edu.cn,
bxu@zju.edu.cn

Abstract

Fault diagnosis of the axial piston pump is critical for maintaining the safety and reliability of modern hydraulic transmission and control systems. Existing deep-learning fault diagnosis models demonstrate effective performance but fail to generalize if new fault patterns occur. Simply fine-tuning these models only with newly collected data leads to the catastrophic forgetting problem, whereas retraining a new fault diagnosis model with the entire historical data is both resource-intensive and time-consuming. To address these issues, a novel class-incremental learning method based on knowledge distillation is proposed for multi-fault diagnosis of the axial piston pump, which can continually learn new fault patterns and preserve fault diagnosis ability on old fault patterns with a limited amount of historical data. To mitigate catastrophic forgetting in subsequent stages, the discrepancy between output-logits of the previous model and that of the current one is introduced in the incremental learning process. Additionally, intermediate feature relationships with different important weights are also employed to further enhance fault diagnosis performance on old fault patterns. Both the comparison experiment and the ablation experiment demonstrate the effectiveness of the proposed method.

Keywords: Axial piston pump, class-incremental learning, fault diagnosis, knowledge distillation

1 Introduction

The axial piston pump serves as a pivotal power component in advanced hydraulic applications, such as aerospace actuators, submarine vessels, and mobile machinery, owing to its superior power density and energy efficiency[1-3]. Ensuring its reliable operation is paramount to maintaining the stability and performance of the whole hydraulic transmission and control system[4]. Nevertheless, uncertain manufacturing and assembling errors, complex structure, and extreme working conditions make the axial piston pump prone to various fault patterns, posing great risks to its reliability and safety in operation[5]. Therefore, multi-fault diagnosis of the axial piston pump is significant for both the reliability-oriented design guidance and the service safety enhancement.

Fault diagnosis techniques for the axial piston pump have witnessed tremendous progress since deep learning enabled data-driven fault pattern recognition from an end-to-end perspective[6-8]. These intelligent fault diagnosis methods typically rely on building a historical dataset and training a class-fixed fault diagnosis model for deployment. However, in practical engineering applications, it is infeasible to gather a comprehensive dataset that contains all possible fault patterns in advance due to the complex structure of the axial piston pump[9]. Consequently, such a class-fixed fault diagnosis model fails to generalize if new fault patterns arise over time. While combining the newly collected data with the whole historical dataset and retraining the fault diagnosis model is theoretically viable, it is time-consuming, resource-demanding, and potentially constrained by data privacy. Conversely, fine-tuning the fault diagnosis model solely with new data leads to

catastrophic forgetting, dramatically degrading the fault diagnosis performance on the previously learned fault patterns[10]. Therefore, it is necessary to develop an effective multi-fault diagnosis method that preserves knowledge of old fault patterns while continuously integrating new fault patterns.

To this end, class-incremental learning, a framework ensuring compatibility with both the old and the new fault patterns, presents a promising solution. Generally, existing incremental learning algorithms can be categorized into architecture-based[11, 12], replay-based[13, 14], parameter regularization-based[15], and knowledge distillation-based methods[16, 17]. Among them, the knowledge distillation paradigm treats the previous fault diagnosis model as the teacher model and initializes a new one as the student model, mitigating catastrophic forgetting by transferring knowledge from the teacher to the student[18]. With minimal or even zero historical data, knowledge distillation-based methods outperform the parameter regularization-based methods on the previously learned fault patterns[19]. Learning without Forgetting (LwF)[16] records the teacher model's output logits on new data as the augmented labels, and minimizes the discrepancy between the student model's output logits and the augmented labels without the engagement of historical data. Incremental Classifier and Representation Learning (iCaRL)[17] also follows a similar idea but selects a fixed amount of prioritized examples from the historical dataset to further retain previous knowledge. Based on iCaRL, a Forward-Back Compatible Representation (FBCR)[20] reserves sufficient embedding space for new patterns to avoid invading previous knowledge. Though these knowledge distillation-based methods alleviate the catastrophic forgetting to some extent through output-layer logits alignment, they overlook the preservation of the intermediate-layer representations that encode discriminative fault features, thereby resulting in suboptimal fault diagnosis performance on old fault patterns.

Therefore, a novel class-incremental learning method based on knowledge distillation is proposed for the multi-fault diagnosis of the axial piston pump. Firstly, a limited number of exemplars are randomly sampled from the historical dataset, and new data are progressively added to form the incremental dataset. Secondly, to tackle the problem of insufficient preservation of old fault patterns, a knowledge distillation loss considering both the output-layer logits and the intermediate-layer representations is specifically designed. Finally, by minimizing the designed loss between the teacher model and the student model, the proposed method achieves good fault diagnosis performance on both the old and new fault patterns.

2 Methodology

The axial piston pump, mainly composed of a shaft, a cylinder block, a valve plate, a retainer, slipper-piston components, etc., is susceptible to multiple fault patterns. In real-world scenarios, the initial training dataset typically covers only a small number of fault patterns. With the operation time going on, new fault patterns that are not presented in the initial training dataset emerge. To learn these new fault patterns, the fault diagnosis model is trained incrementally. The overall training process is split into $T+1$ stages. Specifically, *Stage 0* is the initial training

for a base model, and *Stage 1* ~ *Stage T* is the incremental learning to integrate new fault patterns that occur at each stage, as shown in Figure. 1.

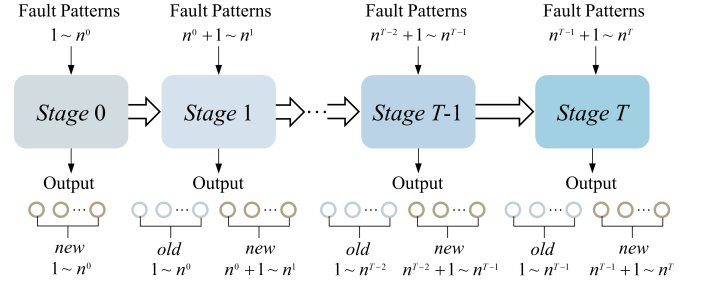


Figure 1: Overall training process for multi-fault diagnosis

2.1 Dataset construction

Ideally, the training dataset $\mathcal{D}_{\text{ideal}}^t$ for the incremental learning *Stage t* ($t = 1, 2, \dots, T$) is

$$\mathcal{D}_{\text{ideal}}^t = \{\underbrace{\mathcal{X}^0, \mathcal{X}^1, \dots, \mathcal{X}^{t-1}}_{\mathcal{H}_{\text{ideal}}^t}, \mathcal{X}^t\} \quad (1)$$

$$\mathcal{X}^j = \left\{ (x_i^j, y_i^j) \right\}_{i=1}^{N^j}, y_i^j \in \mathcal{Y}^j, j = 0, 1, \dots, T \quad (2)$$

where $\mathcal{X}^0$ is the training dataset for the initial training, $\mathcal{X}^1 \sim \mathcal{X}^T$ is the newly collected dataset from the incremental *Stage 1* to *Stage T*, respectively, x_i^j and y_i^j are training samples and true labels, respectively, $\mathcal{Y}^j$ is the label space, if $j \neq j'$, then $\mathcal{Y}^j \cap \mathcal{Y}^{j'} = \emptyset$, meaning that new fault patterns introduced to each stage are distinct and non-repetitive, N^j is the number of training samples contained in $\mathcal{X}^j$. N^j is enough large so that the fault diagnosis model has adequate historical and new training samples to learn both the old and the new fault patterns. For the incremental learning *Stage t*, $\mathcal{X}^0 \sim \mathcal{X}^{t-1}$ forms the ideal historical dataset $\mathcal{H}_{\text{ideal}}^t$, and $\mathcal{X}^t$ is the newly collected dataset only containing new fault patterns emerging at *Stage t*.

However, due to limited storage resources and data privacy, not all the historical data can be recorded in practice. Therefore, the real-world training dataset $\mathcal{D}_{\text{real}}^t$ at the *Stage t* is revised as

$$\mathcal{D}_{\text{real}}^t = \mathcal{H}_{\text{real}}^t \cup \mathcal{X}^t \quad (3)$$

$$\mathcal{H}_{\text{real}}^t = \mathcal{H}_{\text{real}}^{t-1} \cup \left(\mathcal{X}_{\text{sub}}^{t-1} \sim \mathcal{P}(\mathcal{X}^{t-1}, \alpha) \right), \mathbb{E}_p(|\mathcal{X}_{\text{sub}}^{t-1}|) = \alpha N^{t-1} \quad (4)$$

where $\mathcal{H}_{\text{real}}^0 = \emptyset$, $\alpha \in [0, 1)$ is a storage ratio on the historical dataset. Actually, α is a negligible value close to or equal to 0, and $\alpha = 0$ indicates that the fault diagnosis model at the *Stage t* is trained solely on newly collected data. According to (3) and (4), a small subset of exemplars $\mathcal{X}_{\text{sub}}^{t-1}$ is randomly selected from the historical dataset in a storage ratio of α while newly collected data is continuously incorporated to construct the real-world training dataset for the *Stage t*.

2.2 Model architecture

Denote $\mathcal{M}^t$ as the fault diagnosis model consisting of a feature extractor $\mathcal{F}^t$ with L intermediate layers $\{f_i^t; i = 1, 2, \dots, L\}$ and a classifier $\mathcal{C}^t$ at the incremental *Stage t*[21]

$$\mathbf{h}_i^t = f_i^t \circ \dots \circ f_2^t \circ f_1^t(x) \quad (5)$$

$$\hat{\mathbf{y}}^t = \mathcal{M}^t(x) = \mathcal{C}^t(\mathcal{F}^t(x)) = \mathcal{C}^t(\mathbf{h}_L^t) \quad (6)$$

where $\circ$ denotes function composition, $\mathbf{h}_i^t \in \mathbb{R}^{d_i}$ is the i -th intermediate-layer representations, $\hat{\mathbf{y}}^t \in \mathbb{R}^{n'}$ is the output-layer logits, n' is the number of the observed fault patterns until the *Stage t*.

In this article, a one-dimensional convolution neural network (1D-CNN), mainly composed of convolution modules (Conv), batch normalization, activation function, and a two-layer fully connected network (FCN), is employed as the fault diagnosis model. As shown in Figure. 2, all the convolution modules form the feature extractor $\mathcal{F}^t$, and the two-layer FCN forms the classifier $\mathcal{C}^t$. Notably, the detailed architecture of the feature extractor, i.e., the number and the size of the convolution kernels, the convolution strides, and the paddings, remains the same during the incremental learning process. The number of nodes in the input layer of the classifier is unchanged as well, while nodes for new fault patterns are progressively added to the output layer.

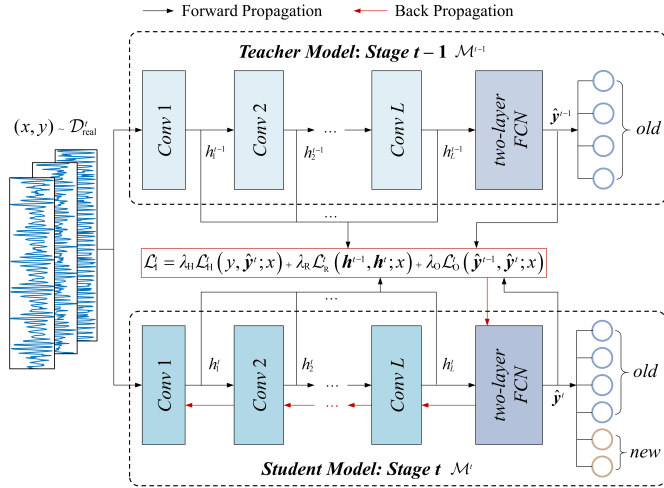


Figure 2: Class-incremental learning based on knowledge distillation for multi-fault diagnosis

2.3 Knowledge distillation loss design

At the incremental learning *Stage t*, the fault diagnosis model $\mathcal{M}^t$ is expected to learn new fault patterns and maintain the capability to diagnose fault patterns learned from *Stage 0* to *Stage t-1*. Generally, given the training dataset $\mathcal{D}_{\text{real}}^t$, the common loss is designed as

$$\mathcal{L}_{\text{H}}(y, \hat{\mathbf{y}}^t; x) = -\sum_{i=1}^{n'} \mathcal{I}(y)_i \log \frac{\exp(\hat{y}_i^t)}{\sum_j \exp(\hat{y}_j^t)} \quad (7)$$

$$\mathcal{I}(y) = [\mathcal{I}(y)_1, \mathcal{I}(y)_2, \dots, \mathcal{I}(y)_i, \dots, \mathcal{I}(y)_{n'}] \quad (8)$$

$$\mathcal{I}(y)_i = \begin{cases} 1, & i = y \\ 0, & i \neq y \end{cases}, i = 1, 2, \dots, n' \quad (9)$$

where $\mathcal{I}(\cdot)$ maps the true label y to a one-hot vector, $\mathcal{I}(y)_i$ is the i -th item in the one-hot vector, and $\hat{y}_i^t$ is the i -th logit in $\hat{\mathbf{y}}^t$.

Simply optimizing (7) enables $\mathcal{M}^t$ to achieve good fault diagnosis performance on new fault patterns but cannot guarantee comparable performance on old fault patterns, probably leading to catastrophic forgetting. Following the teacher-student paradigm of knowledge distillation proposed by Hinton[18], the fault diagnosis model $\mathcal{M}^{t-1}$ which has been well-trained at *Stage t-1*, serves as the teacher model, and $\mathcal{M}^t$ established by adding new nodes at the output layer of the teacher model, serves as the student model. As shown in Figure. 2, the knowledge of the teacher model, i.e., the diagnostic capability for old fault patterns, is transferred to the student model by aligning the output-layer logits and the intermediate-layer representations accordingly.

The loss based on output-layer logits is designed as

$$\mathcal{L}_O(\hat{\mathbf{y}}^{t-1}, \hat{\mathbf{y}}^t; x) = \sum_{i=1}^{n^{t-1}} \tilde{y}_i^{t-1} \log \frac{\tilde{y}_i^{t-1}}{\tilde{y}_i^t} \quad (10)$$

$$\tilde{y}_i^{t-1} = \frac{\exp(\hat{y}_i^{t-1} / \varepsilon)}{\sum_j \exp(\hat{y}_j^{t-1} / \varepsilon)}, \tilde{y}_i^t = \frac{\exp(\hat{y}_i^t / \varepsilon)}{\sum_j \exp(\hat{y}_j^t / \varepsilon)} \quad (11)$$

where ε is a temperature factor to adjust the softness of the output logits distribution.

To further preserve the fault diagnosis performance on previously learned fault patterns, the loss based on intermediate-layer representations that encode discriminative fault features is further introduced

$$\begin{aligned} \mathcal{L}_R(\mathbf{h}^{t-1}, \mathbf{h}^t; x) &= \sum_{i=1}^L \omega_i (1 - \cos(\mathbf{h}_i^{t-1}, \mathbf{h}_i^t)) \\ &= \sum_{i=1}^L \omega_i \left(1 - \frac{\mathbf{h}_i^{t-1} \cdot \mathbf{h}_i^t}{\|\mathbf{h}_i^{t-1}\|_2 \|\mathbf{h}_i^t\|_2} \right) \end{aligned} \quad (12)$$

where $\cos(\cdot)$ denotes cosine similarity, ω_i is a weight to adjust the importance of the i -th intermediate-layer representations, and $\omega_i \propto i$, i.e., representations encoding higher-level features in the deeper intermediate layer are attached more importance.

Therefore, the total knowledge distillation loss is designed as

$$\mathcal{L}_t(y, \hat{\mathbf{y}}^{t-1}, \mathbf{h}^{t-1}, \hat{\mathbf{y}}^t, \mathbf{h}^t; x) = \lambda_H \mathcal{L}_H + \lambda_R \mathcal{L}_R + \lambda_O \mathcal{L}_O, (x, y) \sim \mathcal{D}_{\text{real}}^t \quad (13)$$

where λ_H , λ_R , and λ_O are hyper-parameters that balance the contributions of the three types of loss. By optimizing (13) with appropriate hyper-parameters, the fault diagnosis model can effectively retain diagnostic performance on old fault patterns while learning new fault patterns.

2.4 Class-incremental learning framework

The overall class-incremental learning framework includes initial training at *Stage 0* and class-incremental learning from *Stage 1* to *Stage T*, detailed in Algorithm 1.

At *Stage 0*, the fault diagnosis model $\mathcal{M}^0$ is optimized according to (14) and (15).

$$\mathcal{L}_{\text{H}}^0(y, \hat{\mathbf{y}}^0; x) = -\sum_{i=1}^{n^0} \mathcal{I}(y)_i \log \frac{\exp(\hat{y}_i^0)}{\sum_j \exp(\hat{y}_j^0)}, (x, y) \sim \mathcal{X}^0 \quad (14)$$

$$\arg \min_{\mathcal{M}^0} \mathcal{L}_h^0(y, \hat{y}^0; x) \quad (15)$$

From *Stage 1* to *Stage T*, the fault diagnosis model $\mathcal{M}^t$ ($t = 1, 2, \dots, T$) is optimized according to (7) ~ (13) and (16).

$$\arg \min_{\mathcal{M}^t} \mathcal{L}_i^t(y, \hat{y}^{t-1}, h^{t-1}, \hat{y}^t, h^t; x), t = 1, 2, \dots, T \quad (16)$$

Algorithm 1 Class-incremental learning method based on knowledge distillation

Input: Initial training dataset $\{x_i^0, y_i^0\}_{i=1}^{N^0} \sim \mathcal{X}^0$, incremental stage T , storage ratio α , the number of intermediate layers L , weights ω_i ($i = 1, 2, \dots, L$), temperature factor ε , hyper-parameters λ_H , λ_R , and λ_o , initial training epoch E_0 , incremental learning epoch E_1 , batch size B

- 1: Initialize $\mathcal{M}^0$ according to (5) and (6)
- 2: Define $N = N^0$
- 3: **For** $e = 0, 1, \dots, E_0$
- 4: **For** $b = 0, 1, \dots, N/B-1$
- 5: Calculate $\mathcal{L}_h^0$ according to (6) and (14)
- 6: Optimize $\mathcal{M}^0$ according to (15)
- 7: **End**
- 8: **End**
- 9: **For** $t = 1, 2, \dots, T$
- 10: Initialize $\mathcal{M}^t$ according to (5) and (6)
- 11: Collect $\mathcal{X}^t$ and form $\mathcal{D}_{\text{real}}^t$ according to (3) and (4)
- 12: Update $N = |\mathcal{D}_{\text{real}}^t|$
- 13: **For** $e = 0, 1, \dots, E_1$
- 14: **For** $b = 0, 1, \dots, N/B-1$
- 15: Calculate $\mathcal{L}_i^t$ according to (7) to (13)
- 16: Optimize $\mathcal{M}^t$ according to (16)
- 17: **End**
- 18: **End**
- 19: **End**

Output: well-train $\mathcal{M}^T$

The sampling frequency is 20 kHz and every 1024 consecutive sampling points are considered as a training sample.

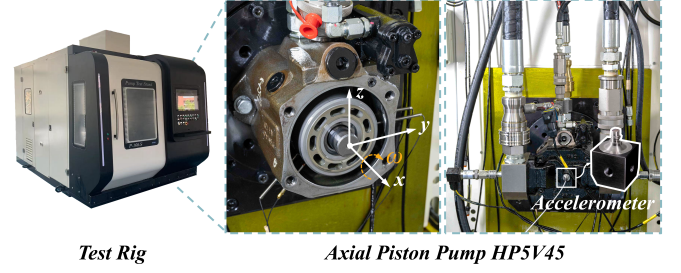


Figure 3: Test rig

In addition to the normal state, four types of fault patterns shown in Figure. 4, including main shaft spline spalling, retainer crack, valve plate scoring, and slipper-piston pair worn, are injected into the tested axial piston pump by replacing the normal components, respectively. Their time-domain vibration signals are depicted in Figure. 5. The fault injection sequence and dataset description for the proposed method are detailed in Table. 1 and Table. 2, respectively.

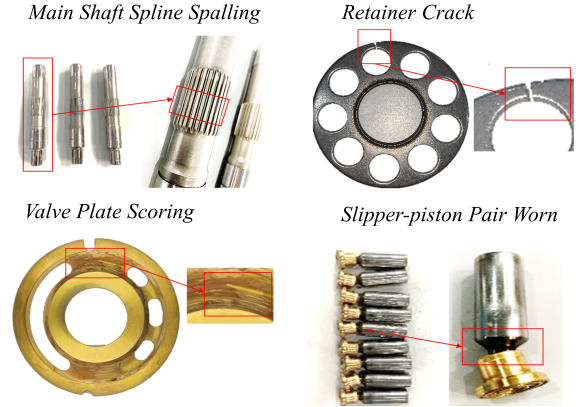


Figure 4: Faulty components

Table 1: Fault injection sequence

Stage	Injected Patterns	Labels
0	Normal State	1
	Main Shaft Spline Spalling	2
1	Retainer Crack	3
2	Valve Plate Scoring	4
3	Slipper-piston Pair Worn	5

Note: During the initial training and incremental learning process, labels are mapped into one-hot vectors according to (8) and (9).

3 Experiment and Result Analysis

3.1 Experimental setting and fault injection

The test rig is shown in Figure. 3. Specifically, the tested axial piston pump is HP5V45 with 9 slipper-piston components, operating under a rated displacement of 45 cm³/r, a rated rotation speed of 2700 rpm, and a rated discharge port pressure of 32 MPa. To monitor periodic vibrations of the tested axial piston pump, a triaxial integrated electronic piezoelectric accelerometer is mounted at the center of its backend cover.

1 – Normal State 2 – Main Shaft Spline Spalling 3 – Retainer Crack 4 – Valve Plate Scoring 5 – Slipper-piston Pair Worn

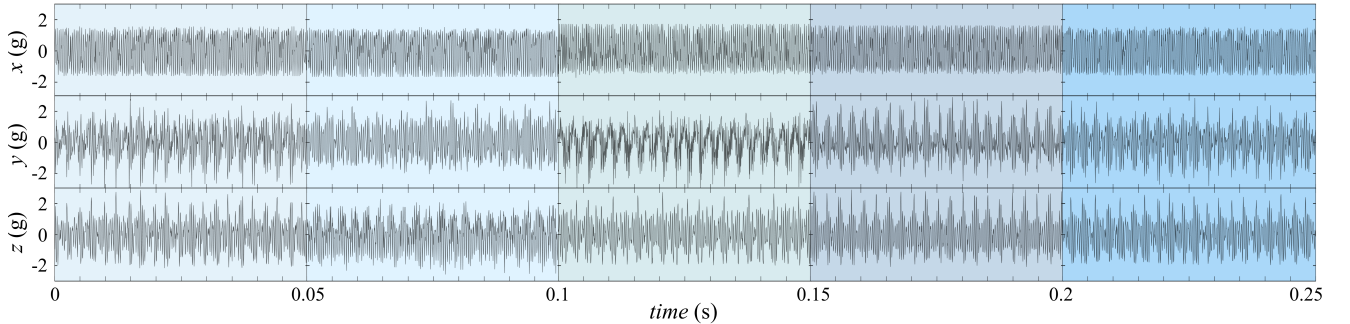


Figure 5: Time-domain vibration signals

Table 2: Dataset description for the proposed method

Stage	Number of Training Samples				Number of Testing Samples	
	$\alpha = 0.005$		$\alpha = 0.01$			
	New	Old	New	Old	New	Old
0	1728	0	1728	0	192	0
1	864	8	864	17	96	192
2	864	12	864	25	96	288
3	864	16	864	33	96	384

Table 3: Detailed architecture of base model

Layer	Kernel Size/Strides	Kernel Number	Input	Output	Padding
Conv 1	64×1/4×1	4	3×1024	4×256	31
Conv 2	4×1/2×1	4	4×256	4×128	1
Conv 3	4×1/2×1	4	4×128	4×64	1
Conv 4	4×1/2×1	4	4×64	4×32	1
FCN	/	/	128	2	/

3.2 Comparison experiment with other methods

The detailed architecture of the base model, i.e., the fault diagnosis model at *Stage 0*, is illustrated in Table. 3. Model parameters in this architecture follow the established 1D-CNN designed for time series input[22]. The input size of *Conv 1* is determined by the number of channels and the length of the input sample. The output size of *FCN* is determined by the number of fault patterns at the current stage. The activation function is LeakyReLU. As mentioned in Section 2.2, the architecture of *Conv 1* ~ *Conv 4* remains the same from *Stage 0* to *Stage 3*, while the number of output nodes in *FCN* increases progressively according to Table. 1.

At *Stage 0*, the optimizer is SGD with a momentum of 0.9 and a weight decay of 10^{-4} . The learning rate is 10^{-3} , the max training epoch is 20, and the batch size is 48. From *Stage 1* to

Stage 3, the proposed method is compared with LwF[16] and fine-tuning. The optimizer of the proposed method is Adam, the batch size is 48, the learning rate is 10^{-2} , the max training epoch is 150, the temperature factor is 10, $[\omega_1, \omega_2, \omega_3, \omega_4] = [0.05, 0.05, 0.2, 0.7]$ and $[\lambda_H, \lambda_R, \lambda_O] = [0.5, 1, 0.5]$. For LwF, the optimizer is Adam, the batch size is 48, the warm-up rate is 10^{-6} and the warm-up epoch is 400, the learning rate is 10^{-4} , and the max training epoch is 200. For fine-tuning, the optimizer is Adam, the batch size is 48, the learning rate is 10^{-2} , and the max training epoch is 150. These training settings are manually configured to ensure that all the methods are trained to converge and reach a training accuracy of 100% for fair comparison.

To evaluate the effectiveness of each method, three performance metrics are defined as follows

$$\mathcal{A}_{old}^i = \frac{C_{old}^i}{N_{old}^i} \times 100\%, i = 1, 2, 3 \quad (17)$$

$$\mathcal{A}_{new}^i = \frac{C_{new}^i}{N_{new}^i} \times 100\%, i = 0, 1, 2, 3 \quad (18)$$

$$\mathcal{A}_{avg}^i = \frac{C_{new}^i + C_{old}^i}{N_{new}^i + N_{old}^i} \times 100\%, i = 0, 1, 2, 3 \quad (19)$$

where C_{old}^i and C_{new}^i are the number of correctly diagnosed testing samples of old fault patterns and new fault patterns at *Stage i*, respectively, N_{old}^i and N_{new}^i are the number of testing samples of old fault patterns and new fault patterns at *Stage i*, respectively, $\mathcal{A}_{old}^i$ indicates the fault diagnosis performance on old fault patterns at *Stage i*, $\mathcal{A}_{new}^i$ indicates the fault diagnosis performance on new fault patterns at *Stage i*, and $\mathcal{A}_{avg}^i$ indicates the comprehensive fault diagnosis performance on all fault patterns at *Stage i*. All the methods are trained to converge. The training process is repeated 5 times, and the average performance metrics are considered as the fault diagnosis performance. The comparison results are listed in Table. 4.

Table 4: Comparison experiment results (%)

Stage		Proposed Method ($\alpha = 0.005$)	Proposed Method ($\alpha = 0.01$)	LwF	Fine-tuning
0	$\mathcal{A}_{new}^0$	100	100	100	100
	$\mathcal{A}_{avg}^0$	100	100	100	100
1	$\mathcal{A}_{old}^1$	55.10±8.28	90.52±11.57	12.08±11.30	0
	$\mathcal{A}_{new}^1$	100	100	99.58±0.83	100
	$\mathcal{A}_{avg}^1$	70.01±5.52	93.08±7.71	41.25±7.58	33.33
2	$\mathcal{A}_{old}^2$	37.01±14.04	69.44±11.02	7.57±3.52	0
	$\mathcal{A}_{new}^2$	100	100	97.08±1.38	100
	$\mathcal{A}_{avg}^2$	52.76±10.53	77.08±8.27	29.95±2.67	25
3	$\mathcal{A}_{old}^3$	37.86±7.03	69.06±4.25	6.56±5.53	0
	$\mathcal{A}_{new}^3$	100	100	97.08±1.21	100
	$\mathcal{A}_{avg}^3$	50.29±5.62	75.25±3.40	24.67±4.53	20

For new fault patterns, all the methods demonstrate superior fault diagnosis performance at each stage, where the accuracy of LwF is slightly lower than 100%, and that of the proposed method and fine-tuning is 100%.

In contrast, for old fault patterns, all the methods suffer from different degrees of decline in accuracy, as shown in the Figure. 6. Particularly, the accuracy of fine-tuning drops directly to 0% at each stage, indicating its poor ability in maintaining fault diagnosis performance on old fault patterns. LwF outperforms fine-tuning, with an accuracy of $6.56 \pm 5.53\%$ on old fault patterns at *Stage 3*. Given a small storage ratio, the proposed method surpasses LwF and fine-tuning. As the storage ratio increases from 0.005 to 0.01, the accuracy of the proposed method on old fault patterns and all fault patterns increases by 31.20% and 24.96%, respectively, at *Stage 3*. This highlights the significance of leveraging historical data in the incremental training process.

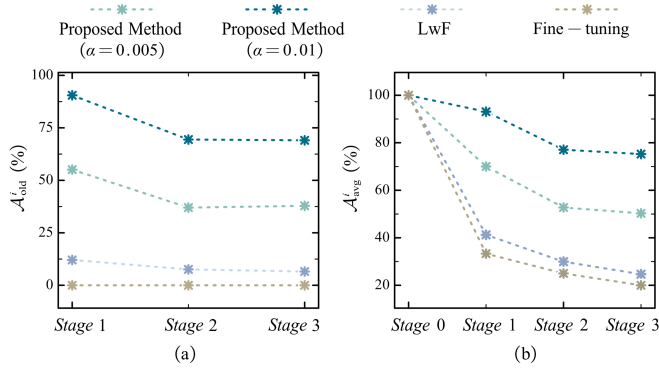


Figure 6: Comparison regarding A_{ol}^i and A_{avg}^i

Furthermore, the recalls are calculated at *Stage 3* to analyze the degradation of the fault diagnosis performance for each fault pattern

$$\mathcal{R}^i = \frac{C^i}{N^i} \times 100\%, i = 1, 2, 3, 4, 5 \quad (20)$$

where N^1 , N^2 , N^3 , N^4 , and N^5 are the number of testing samples of the normal state, main shaft spline spalling, retainer crack, valve plate scoring, and slipper-piston pair worn at *Stage 3*, respectively, C^1 , C^2 , C^3 , C^4 , and C^5 are the correctly diagnosed testing samples of the normal state, main shaft spline spalling, retainer crack, valve plate scoring, and slipper-piston pair worn at *Stage 3*, respectively, $\mathcal{R}^1$, $\mathcal{R}^2$, $\mathcal{R}^3$, $\mathcal{R}^4$, and $\mathcal{R}^5$ are the recall of the normal state, main shaft spline spalling, retainer crack, valve plate scoring, and slipper-piston pair worn at *Stage 3*, respectively. $\mathcal{R}^i$ indicates fault diagnosis performance on each fault pattern at *Stage 3*. The

average recalls at *Stage 3* are listed in Table. 5 and visualized in Figure. 7. The results generally demonstrate a common trend among all the methods, i.e., as the incremental learning progresses, fault diagnosis performance degrades more severely on older fault patterns compared to new ones, suggesting the catastrophic forgetting phenomenon. However, two exceptions are observed in the proposed method, i.e., $\mathcal{R}^2$ exceeds $\mathcal{R}^3$ and $\mathcal{R}^4$ at a storage ratio of 0.005, and $\mathcal{R}^4$ falls below both $\mathcal{R}^2$ and $\mathcal{R}^3$ at a storage ratio of 0.01.

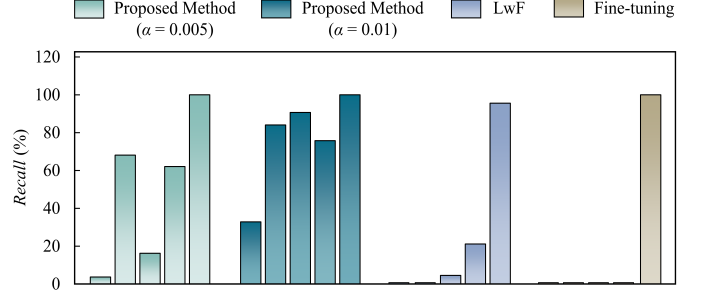


Figure 7: Recalls of injected patterns at *Stage 3*

3.3 Ablation experiment of the proposed method

To verify the contribution of each type of loss in (13), the ablation experiment is conducted. Specifically, Ablation 1 sets $[\lambda_H, \lambda_R, \lambda_O] = [1, 0, 0]$ and Ablation 2 sets $[\lambda_H, \lambda_R, \lambda_O] = [0.5, 0, 0.5]$. Other training settings are consistent with those described in Section 3.2. The ablation experiment results are listed in Table. 5, and results regarding A_{ol}^i and A_{avg}^i are drawn in Figure. 8.

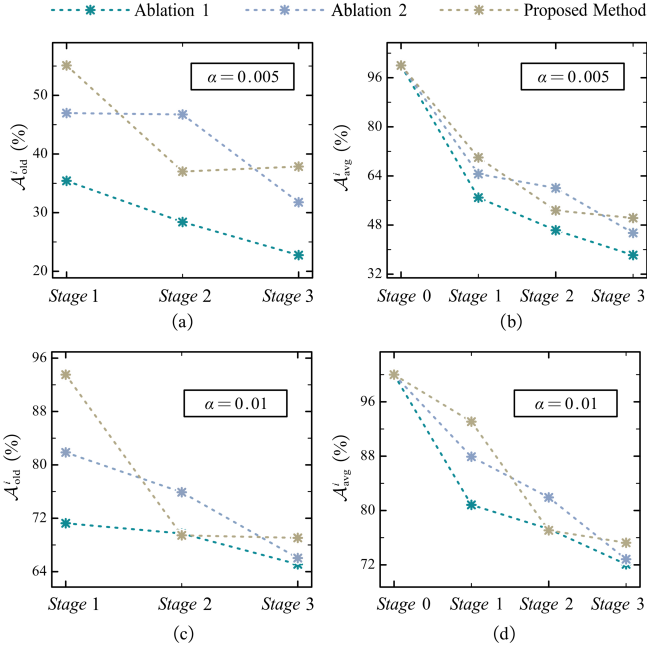
Similarly, all the methods achieve an accuracy of 100% on new fault patterns and generally degrade on old fault patterns during the incremental training process. At a storage ratio of 0.005, ablation 2 and the proposed method always outperform ablation 1, indicating that aligning the output-logits and intermediate-representations enhances the capability to preserve fault diagnosis performance on old fault patterns. According to Figure. 8(a) and 8(c), the proposed method performs worse than Ablation 2 on old fault patterns at *Stage 2*. However, from *Stage 2* to *Stage 3*, the accuracy of Ablation 2 on old fault patterns decreases by 9.86% and 14.97%, at a storage ratio of 0.01 and 0.005, respectively. While that of the proposed method only decreases by 0.38% at a storage ratio of 0.01, and even increases by 0.85% at a storage ratio of 0.005. This indicates that, compared to relying solely on output-layer logits, joint alignment of both output-layer logits and the intermediate-layer representations further improves the ability to retain fault diagnosis performance on old fault patterns.

Table 5: Recalls of injected patterns at *Stage 3* (%)

Recall	Proposed Method ($\alpha = 0.005$)	Proposed Method ($\alpha = 0.01$)	LwF	Fine-tuning
$\mathcal{R}^1$	3.62	32.79	0	0
$\mathcal{R}^2$	68.04	84	0	0
$\mathcal{R}^3$	16.17	90.67	4.46	0
$\mathcal{R}^4$	62.04	75.71	21.13	0
$\mathcal{R}^5$	100	100	95.50	100

Table 6: Ablation experiment results (%)

Stage		$\alpha = 0.005$			$\alpha = 0.01$		
		Ablation 1	Ablation 2	Proposed Method	Ablation 1	Ablation 2	Proposed Method
0	$\mathcal{A}_{\text{new}}^0$	100	100	100	100	100	100
	$\mathcal{A}_{\text{avg}}^0$	100	100	100	100	100	100
1	$\mathcal{A}_{\text{old}}^1$	35.42 ± 21.49	46.98 ± 10.62	55.10 ± 8.28	71.25 ± 16.25	81.87 ± 13.78	90.52 ± 11.57
	$\mathcal{A}_{\text{new}}^1$	100	100	100	100	100	100
	$\mathcal{A}_{\text{avg}}^1$	56.94 ± 14.33	64.65 ± 7.08	70.01 ± 5.52	80.83 ± 10.84	87.92 ± 9.18	93.08 ± 7.71
2	$\mathcal{A}_{\text{old}}^2$	28.40 ± 11.48	46.74 ± 8.35	37.01 ± 14.04	69.72 ± 8.51	75.90 ± 9.75	69.44 ± 11.02
	$\mathcal{A}_{\text{new}}^2$	100	100	100	100	100	100
	$\mathcal{A}_{\text{avg}}^2$	46.30 ± 8.61	60.05 ± 6.26	52.76 ± 10.53	77.29 ± 6.39	81.93 ± 7.31	77.08 ± 8.27
3	$\mathcal{A}_{\text{old}}^3$	22.76 ± 8.97	31.77 ± 4.29	37.86 ± 7.03	65.10 ± 3.40	66.04 ± 6.78	69.06 ± 4.25
	$\mathcal{A}_{\text{new}}^3$	100	100	100	100	100	100
	$\mathcal{A}_{\text{avg}}^3$	38.21 ± 7.18	45.42 ± 3.43	50.29 ± 5.62	72.08 ± 2.72	72.83 ± 5.42	75.25 ± 3.40

Figure 8: Ablation regarding $\mathcal{A}_{\text{old}}^i$ and $\mathcal{A}_{\text{avg}}^i$

4 Conclusions

For multi-fault diagnosis of the axial piston pump, a class-incremental learning method based on knowledge distillation is proposed to integrate newly emerging fault patterns as well as preserving fault diagnosis performance on old fault patterns. Comparison experiment with LwF and fine-tuning, and ablation experiment around each loss item are conducted to verify the effectiveness of the proposed method. Main conclusions are drawn as follows.

(1) Storing only a small portion of historical data can greatly improve the fault diagnosis performance on old fault patterns. At Stage 3, increasing the storage ratio from 0.005 to 0.01 boosts the accuracy of the proposed method on old fault patterns by 31.20%.

(2) Joint alignment of the output-layer logits and the intermediate-layer representations enhances knowledge preservation on old fault patterns. Adding the loss item regarding the output-layer logits boosts the accuracy of the proposed method on old fault patterns by 9.01% ($\alpha = 0.005$) and 0.94% ($\alpha = 0.01$), respectively. Further adding the loss item regarding the intermediate-layer representations boosts the accuracy of the proposed method on old fault patterns by 6.09% ($\alpha = 0.005$) and 3.02% ($\alpha = 0.01$), respectively.

There is still room for further discussions and enhancements.

(1) Theoretically, the proposed method can simultaneously learn several fault patterns at every incremental learning stage, but the case of concurrent faults remains to be empirically explored due to the limited types of fault patterns introduced in this article. More fault patterns of the axial piston pump need to be considered and collected in future work.

(2) Hyper-parameters in this article are given manually and may probably lead to a suboptimal performance in real-world scenarios when operating condition changes frequently. Therefore, how to set appropriate hyper-parameters adaptive to different pumps and operating conditions needs to be analyzed to achieve a more generalized fault diagnosis performance.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (U24B2049).

References

- [1] S Mukherjee, L Shang, and A Vacca. Numerical analysis and experimental validation of the coupled thermal effects in swashplate type axial piston machines. *Mechanical Systems and Signal Processing*, 220:111673, 2024.
- [2] P Michael, K Stelson, D Williams, and H Malik. Dynamometer testing of hydraulic fluids in an axial piston

- pump under simulated backhoe loader trenching conditions. In *Fluid Power Systems Technology* (American Society of Mechanical Engineers), 86335:V001T01A014, 2022.
- [3] L V Larsson and P Krus. Modelling of the swash plate control actuator in an axial piston pump for a hardware-in-the-loop simulation test rig. In *Fluid Power Systems Technology* (American Society of Mechanical Engineers), 50473:V001T01A044, 2016.
- [4] I Baus, R Rahmfeld, A Schumacher, and H C Pedersen. Systematic methodology for reliability analysis of components in axial piston units. In *Fluid Power Systems Technology* (American Society of Mechanical Engineers), 59339:V001T01A009, 2019.
- [5] R Ivantysyn and J Weber. Advancing thermal monitoring in axial piston pumps: Simulation, measurement, and boundary condition analysis for efficiency enhancement. *International Journal of Fluid Power*, 547-590, 2024.
- [6] S Wang, J Xiang, Y Zhong, and H Tang. A data indicator-based deep belief networks to detect multiple faults in axial piston pumps. *Mechanical Systems and Signal Processing*, 112:154-170, 2018.
- [7] Y He, H Tang, Y Ren, and A Kumar. A semi-supervised fault diagnosis method for axial piston pump bearings based on DCGAN. *Measurement Science and Technology*, 32(12):125104, 2021.
- [8] S Wang et al. Few-shot fault diagnosis of axial piston pump based on prior knowledge-embedded meta learning vision transformer under variable operating conditions. *Expert Systems with Applications*, 269:126452, 2025.
- [9] S Liu, J Zhang, W Huang, F Lyu, D Wang, and B Xu. Temporal-spatial attention network: A novel axial piston pump coupled fault diagnosis method. *IEEE Transactions on Instrumentation and Measurement*, 73:1-15, 2024.
- [10] E Belouadah, A Popescu, and I Kanellos. A comprehensive study of class incremental learning algorithms for visual tasks. *Neural Networks*, 135:38-54, 2021.
- [11] A A Rusu et al. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.
- [12] Y-X Wang, D Ramanan, and M Hebert. Growing a brain: Fine-tuning by increasing model capacity. Proceedings of *the IEEE Conference on Computer Vision and Pattern Recognition*, 2471-2480, 2017.
- [13] R Aljundi, M Lin, B Goujaud, and Y Bengio. Gradient based sample selection for online continual learning. *Advances in Neural Information Processing Systems*, 32, 2019.
- [14] A Odena, C Olah, and J Shlens. Conditional image synthesis with auxiliary classifier gans. *International Conference on Machine Learning*, 2642-2651, 2017.
- [15] J Kirkpatrick et al. Overcoming catastrophic forgetting in neural networks. Proceedings of *the National Academy of Sciences*, 114(13):3521-3526, 2017.
- [16] Z Li and D Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935-2947, 2017.
- [17] S-A Rebuffi, A Kolesnikov, G Sperl, and C H Lampert. icarl: Incremental classifier and representation learning. Proceedings of *the IEEE Conference on Computer Vision and Pattern Recognition*, 2001-2010, 2017.
- [18] G Hinton, O Vinyals, and J Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.
- [19] G M Van de Ven, T Tuytelaars, and A S Tolias. Three types of incremental learning. *Nature Machine Intelligence*, 4(12):1185-1197, 2022.
- [20] S Yan, H Shao, X Wang, and J Wang. Few-shot class-incremental learning for system-level fault diagnosis of wind turbine. *IEEE/ASME Transactions on Mechatronics*, 2024.
- [21] M Kang, J Park, and B Han. Class-incremental learning by knowledge distillation with adaptive feature consolidation. Proceedings of *the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16071-16080, 2022.
- [22] M. Ji, et al. A neural network compression method based on knowledge-distillation and parameter quantization for the bearing fault diagnosis. *Applied Soft Computing*, 127:109331, 2022.