Autoencoder-Based Discriminant Locality-Preserving Projections for Fault Diagnosis

Ruisheng Ran (冉瑞生), *Member, IEEE*, Ting Wang (王婷), Wenfeng Zhang (张文锋), *Member, IEEE*, and Bing Fang (房斌), *Senior Member, IEEE*

***Abstract*— In complex industrial production environments, the efficacy of fault diagnostic techniques has become increasingly important and can enhance the reliability and safety of systems. In recent years, the discriminant locality-preserving projection (DLPP) algorithm has shown significant effectiveness in extracting meaningful features from complex industrial data. However, DLPP involves only the operation of projecting high-dimensional data onto a lower dimensional space, which is a one-way mapping process and lacks the verification of whether the low-dimensional data projected can accurately and effectively represent the original data. This might result in the loss of vital information in the original data, consequently limiting the performance of DLPP. In this article, we introduce a novel DLPP approach denoted as autoencoder-based DLPP (DLPP-AE), which is predicated upon the autoencoder. DLPP-AE establishes a bidirectional mapping process: in the encoding stage, DLPP serves as a mapping mechanism that transforms the initial high-dimensional data into a low-dimensional embedded representation; whereas in the decoding stage, the low-dimensional embedded data generated in the encoding stage are remapped back to its original high-dimensional form. This effectively resolves the issue of DLPP's inability to perform reverse validation on low-dimensional embeddings. To evaluate the performance of the proposed method, we conducted a comprehensive case study using three laboratory datasets and one real industrial dataset. The experimental results confirm the superior fault diagnosis capability of the DLPP-AE method.**

***Index Terms*— Autoencoder, discriminant locality-preserving projections (DLPPs), fault diagnosis, feature extraction, manifold learning.**

I. Introduction

THE widespread use of sensing devices, measurement instruments, and advanced control equipment has increased the complexity and level of automation in modern industries [1]. However, this has also increased the risk of failures [2]. The components or subsystems within these systems are interconnected, and many critical parts, such as bearings and gears, are highly vulnerable. Once a failure occurs, it can impact various aspects of the system, resulting in a range of consequences. This may lead to substantial economic losses or even casualties [3]. Consequently, efficient fault diagnosis techniques are critical in modern industries [4], [5].

To address these challenges, researchers have developed numerous fault diagnosis methods, which can be categorized from a technological perspective into three main types: 1) model-driven methods; 2) knowledge-driven methods; and 3) data-driven methods [6]. Model-driven fault diagnosis methods are grounded in fundamental system principles, utilizing precise mathematical models to generate characteristic information that reflects the system's state for detection and diagnosis [7]. This approach is particularly well-suited for systems characterized by a limited number of inputs and outputs, in which the relationships between variables are relatively straightforward. Knowledge-based fault diagnosis methods are founded on extensive domain expertise, encompassing the relationships between inputs and outputs, causal links among variables, and the system's design logic and decision-making mechanisms [8]. These methods involve the development of diagnostic models and reasoning frameworks to analyze system faults. Data-driven fault diagnosis methods extract features from large volumes of industrial process data using data mining techniques and construct models for fault classification employing methods such as machine learning [9]. These methods emphasize two key components: data and models. With the rapid development of industry and the emergence of big data, model-driven methods and knowledge-driven methods have become increasingly less applicable [10]. In contrast, data-driven methods do not necessitate precise mathematical models or extensive knowledge of system mechanisms. Instead, they rely exclusively on process data to model and diagnose faults. Consequently, data-driven fault diagnosis has emerged as a significant research focus [11].

The primary challenge faced by data-driven fault diagnosis methods is how to extract more meaningful information from vast amounts of process data characterized by high dimensionality, nonlinearity, and strong interdependencies. Fortunately, dimensionality reduction (DR) techniques have shown excellent performance in handling such high-dimensional data. Therefore, the application of DR techniques for feature extraction has significantly increased in the field of fault

Received 23 December 2024; revised 18 February 2025; accepted 21 February 2025. Date of publication 17 March 2025; date of current version 8 April 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62376042, in part by the Key Project for Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJZD-K202100505, in part by the Natural Science Foundation of Chongqing under Grant CSTB2024NSCQ-KJFZZDX0036, and in part by the Key Special Project for Technological Innovation and Application Development in Chongqing under Grant CSTB2023TIAD-KPX0050 and Grant CSTB2022TIAD-KPX0176. The Associate Editor coordinating the review process was Dr. Arunava Naha. *(Corresponding author: Wenfeng Zhang.)*

Ruisheng Ran, Ting Wang, and Wenfeng Zhang are with the College of Computer and Information Science, College of Intelligent Science, Chongqing Normal University, Chongqing 401331, China (e-mail: rshran@cqnu.edu.cn; 2443610915@qq.com; itzhangwf@cqnu.edu.cn).

Bing Fang is with the College of Computer Science, Chongqing University, Chongqing 400044, China (e-mail: fb@cqu.edu.cn).

Digital Object Identifier 10.1109/TIM.2025.3551906

diagnosis [12]. Some classical DR algorithms, such as principal component analysis (PCA) [13], Fisher discriminant analysis (FDA) [14], and partial least squares (PLS) [15], are commonly applied to fault diagnosis. PCA, being the most prevalent DR algorithm, is utilized to eliminate redundant information. In order to further address relationships between different data categories, FDA has been proposed. PLS addresses the issue overlooked by PCA, considering the impact of input on output variations. Although PCA, FDA, and FLS and their variants are widely used, they overlook the manifold structure and neighborhood information of the data. This oversight can result in a decrease in the performance of feature extraction.

Fortunately, manifold learning methods can preserve the local similarity of data while reducing its dimensionality. Due to their outstanding feature extraction performance, manifold learning methods have found widespread applications in pattern recognition and fault diagnosis [16]. For example, Li et al. proposed a rolling bearing fault diagnosis method based on the locally linear embedding (LLE) technique [17]. This approach replaces the Euclidean distance with the Mahalanobis distance to enhance fault recognition accuracy. Zhang and Shang [18] introduced two novel distance-based metric methods—distance correlation entropy (DCE) and ordinal distance complexity measure (ODCM)—to replace the traditional metrics in the uniform manifold approximation and projection (UMAP) algorithm, applying them to fault diagnosis. Wang et al. [19] proposed a multimodal process monitoring method based on dynamic locality-preserving PCA. By integrating locality-preserving projection (LPP) with PCA, this approach enhances the effectiveness of multimodal fault detection.

While these manifold learning methods effectively preserve the local similarity of the data after DR, they do not consider discriminative information. Discriminative information refers to the information used to distinguish differences in categories or states between data. This oversight could lead to the loss of discriminative information, resulting in a decrease in classification performance. To tackle this problem, certain refined manifold learning algorithms, exemplified by discriminative LPP (DLPP) [20], consider discriminative information when reducing high-dimensional data. Although DLPP considers discriminative information during DR, its efficacy is impacted by the small sample size (SSS) problem. Lu et al. [21] introduced a DLPP algorithm based on the maximum margin criterion (DLPP/MMC). DLPP/MMC changes the ratio between the maximization of interclass scatter and intraclass scatter to the maximization of the difference between the two scatters, addressing the SSS problem of DLPP.

However, DLPP and its variants encounter a fundamental challenge; they involve only the operation of projecting high-dimensional data onto a lower dimensional space, which is a one-way mapping process. However, it lacks the verification of whether the low-dimensional data projected can accurately and effectively represent the original data. This might result in the loss of vital information in the original data, consequently limiting the performance of DLPP.

Recently, deep learning and representation learning have achieved success in many application domains [22], [23], [24]. Among these, autoencoders are commonly used as the foundation for initializing neural networks due to their powerful capability to handle complex nonlinear data.

Inspired by the "encoding–decoding" structure of autoencoder, this article introduces a novel DLPP algorithm named autoencoder-based DLPP (DLPP-AE). Built upon the foundation of a linear autoencoder structure, DLPP-AE constructs a bidirectional mapping; in the encoding stage, DLPP serves as a mapping mechanism that transforms the initial high-dimensional data into a low-dimensional embedded representation; whereas in the decoding stage, the low-dimensional embedded data generated in the encoding stage are remapped back to its original high-dimensional form. This enables data points in the latent space to be reconstructed back to their original form, effectively addressing the limitation of DLPP in performing reverse validation of low-dimensional embeddings. This bidirectional mapping can preserve both the neighborhood information and discriminative properties of the original data through the DLPP/MMC during the encoding stage, while also ensuring that the obtained low-dimensional data more accurately represent the original data through the reconstruction mechanism of autoencoders during the decoding stage.

After using DLPP-AE for fault feature extraction, a K nearest neighbor (KNN) classifier is employed to categorize the extracted fault features for industry process fault diagnosis. During feature extraction with DR, a commonly utilized criterion, the Akaike information criterion (AIC), is employed to ascertain the optimal reduction order. To evaluate the performance of the proposed method, we conducted a comprehensive case study using three laboratory datasets and one real industrial dataset. The experimental results confirm the superior fault diagnosis capability of the DLPP-AE method. Extensive experiments show that the proposed DLPP-AE method has a high feature extraction ability and a high generalization ability.

The contributions of this article can be summarized as follows.

1) The proposed DLPP-AE integrates the advantages of DLPP and autoencoder, ensuring that the low-dimensional embedding preserves the local neighborhood information of the original data points while ensuring that the low-dimensional embedding more accurately represents complex high-dimensional data through the decoding stage, thus solving the problem of information loss caused by the one-way mapping of DLPP.
2) Although DLPP-AE is a linear, single-layer structure, its performance is superior to some of the latest deep neural network methods, demonstrating the effectiveness of this approach. Hence, this structure is expected to be extended as a basic paradigm to other feature extraction methods.
3) In industrial fault diagnosis scenarios, the proposed DLPP-AE method demonstrates superior fault diagnosis capability compared to traditional machine learning methods and some recent deep learning methods. This method exhibits exceptional performance in fault

diagnosis and holds the potential for widespread practical implementation.

The subsequent sections of this article are structured as follows. Section II offers a concise review of the DLPP method and autoencoder. Section III introduces the new DLPP method DLPP-AE for fault diagnosis. Section IV conducts experimental studies using four fault datasets to validate the performance of the proposed method. Finally, Section V concludes this article.

II. Review of Related Works

A. Discriminant Locality-Preserving Projections

The primary objective of the DLPP algorithm is to find the optimal projection matrix $\boldsymbol{A}$, which preserves the discriminative and local similarity properties of the original data after projection.

A given set of sample collection $\{\boldsymbol{x}_i\}$ can be represented as an $M \times N$ matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N]$, where M and N denote the dimensions and the number of samples, respectively. Each $\boldsymbol{x}_i$ belongs to a sample class $\{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_C\}$. $\boldsymbol{x}_i$ is mapped to $\boldsymbol{y}_i$ through matrix $\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_d] \in \mathbb{R}^{M\times d}$, denoted as $\boldsymbol{y}_i = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{x}_i$. The objective of DLPP is to maximize the following function:

$$L(\boldsymbol{A}) = \frac{\boldsymbol{A}^{\mathrm{T}}\boldsymbol{F}\boldsymbol{H}\boldsymbol{F}^{\mathrm{T}}\boldsymbol{A}}{\boldsymbol{A}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{A}}. \tag{1}$$

Here, $\boldsymbol{L}$ and $\boldsymbol{H}$ are the Laplacian matrices. $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, where $\boldsymbol{W}$ is the within-class weight matrix, $\boldsymbol{W}_{ij}^{C} = \exp(-\|\boldsymbol{x}_i^{C} - \boldsymbol{x}_j^{C}\|^2/t)$ is the weight between $\boldsymbol{y}_i^{C}$ and $\boldsymbol{y}_j^{C}$, $\boldsymbol{D} = \mathrm{diag}\{\boldsymbol{D}^1, \boldsymbol{D}^2, \ldots, \boldsymbol{D}^C\}$, and $\boldsymbol{D}_{ii}^{k} = \sum_i \boldsymbol{W}_{ij}^{k}$. $\boldsymbol{H} = \boldsymbol{E} - \boldsymbol{B}$, where $\boldsymbol{B}$ is the between-class weight matrix, $\boldsymbol{B}_{ij} = \exp(-\|\boldsymbol{f}_i - \boldsymbol{f}_j\|^2/t)$ is the weight between the ith class mean $\{\boldsymbol{f}_i\}$ and the ith class mean $\{\boldsymbol{f}_j\}$, and t is an empirical parameter. $\boldsymbol{E}$ is a diagonal matrix, and its elements are the sums of the columns (or rows) of matrix $\boldsymbol{B}$, $\boldsymbol{E}_{ii} = \sum_i \boldsymbol{B}_{ij}$. And the matrix $\boldsymbol{F} = \left[\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_C\right]$.

Lu et al. [21] introduced a new DLPP approach called DLPP/MMC. DLPP/MMC changes the ratio between the maximization of within-class scatter and between-class scatter to the maximization of the difference between the two scatters, addressing the SSS problem of DLPP. Define local within-class scatter $\boldsymbol{S}_w^{L} = \boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^{\mathrm{T}}$ and local between-class scatter $\boldsymbol{S}_b^{L} = \boldsymbol{F}\boldsymbol{H}\boldsymbol{F}^{\mathrm{T}}$. The scatter matrices of the projected training sample matrix $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{X}$ are the between-class scatter matrix $\tilde{\boldsymbol{S}}_b^{L} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_b^{L}\boldsymbol{A}$ and the within-class scatter matrix $\tilde{\boldsymbol{S}}_w^{L} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{S}_w^{L}\boldsymbol{A}$. The objective function of DLPP/MMC is given by

$$\begin{aligned} J(\boldsymbol{A}) &= \mathrm{tr}\left(\tilde{\boldsymbol{S}}_b^{L} - \alpha\tilde{\boldsymbol{S}}_w^{L}\right) \\ &= \mathrm{tr}\left(\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{S}_b^{L} - \alpha\boldsymbol{S}_w^{L}\right)\boldsymbol{A}\right) \\ &= \mathrm{tr}\left(\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{F}\boldsymbol{H}\boldsymbol{F}^{\mathrm{T}} - \alpha\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^{\mathrm{T}}\right)A\right). \end{aligned} \tag{2}$$

The optimization algorithm described above can be solved by finding the optimal solution through the generalized eigenvalue, where $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_d$ is the eigenvector corresponding to the d largest eigenvalues of the matrix $(\boldsymbol{S}_b^{L} - \alpha\boldsymbol{S}_w^{L})$.

B. Autoencoder

An autoencoder is a standard encoder–decoder structure comprising input, hidden, and output layers. The encoder consists of an input layer and a hidden layer, responsible for encoding input data from the original domain into the feature domain. The decoder consists of a hidden layer and an output layer, responsible for decoding encoded data from the feature domain back to the original domain. This encoding–decoding process in the autoencoder aims to reconstruct the original data and extract more effective features from it. Training models using reconstructed feature data can lead to improved classification performance. Autoencoders have been a research focus, and various improvements based on autoencoders have been proposed, including stacked autoencoders, stacked sparse autoencoders, denoising autoencoders, and deep generative Laplacian autoencoders. Each layer of the encoder and decoder is linear, and autoencoders without nonlinear activation functions are referred to as linear autoencoders. The new method proposed in this article is based on a linear autoencoder.

III. Proposed Methodology

The methodology introduced in this article builds upon DLPP. To address the SSS problem of the original DLPP, a variant called DLPP/MMC is employed. The DLPP/MMC algorithm maximizes the difference between the local preservation of interclass scatter and intraclass scatter, finding an optimal projection subspace for classification. This resolves the SSS problem while retaining the original high-dimensional data manifold structure and discriminative properties of DLPP. However, this projection algorithm is unidirectional, lacking the ability to reverse-validate whether low-dimensional embeddings accurately depict the original high-dimensional data. In order to address this issue, a novel DLPP-AE method based on a linear autoencoder is proposed in this article. The framework of DLPP-AE is presented, followed by the construction and optimization of the objective function. Finally, the application of the DLPP-AE-based fault diagnosis model is elaborated.

A. Framework and the Objective Function

Using an encoder–decoder architecture, DLPP-AE establishes a bidirectional mapping. In the encoding stage, DLPP/MMC serves as a mapping mechanism that transforms the initial high-dimensional data into a low-dimensional embedded representation while preserving the discriminative and local similarity properties of the data. In the decoding stage, the low-dimensional embedded data generated in the encoding stage is remapped back to its original high-dimensional form. This remapping process ensures that the embedded data closely match the "semantic" of the original data, thereby avoiding the loss of important information from the original data. Since the new DLPP method operates within the encoder–decoder framework, it is aptly named DLPP-AE. The schematic representation of the DLPP-AE methodology is delineated in Fig. 1.

Assume we have n original samples $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N]$ in the space $\mathbb{R}^M$. Initially, we construct a neighborhood

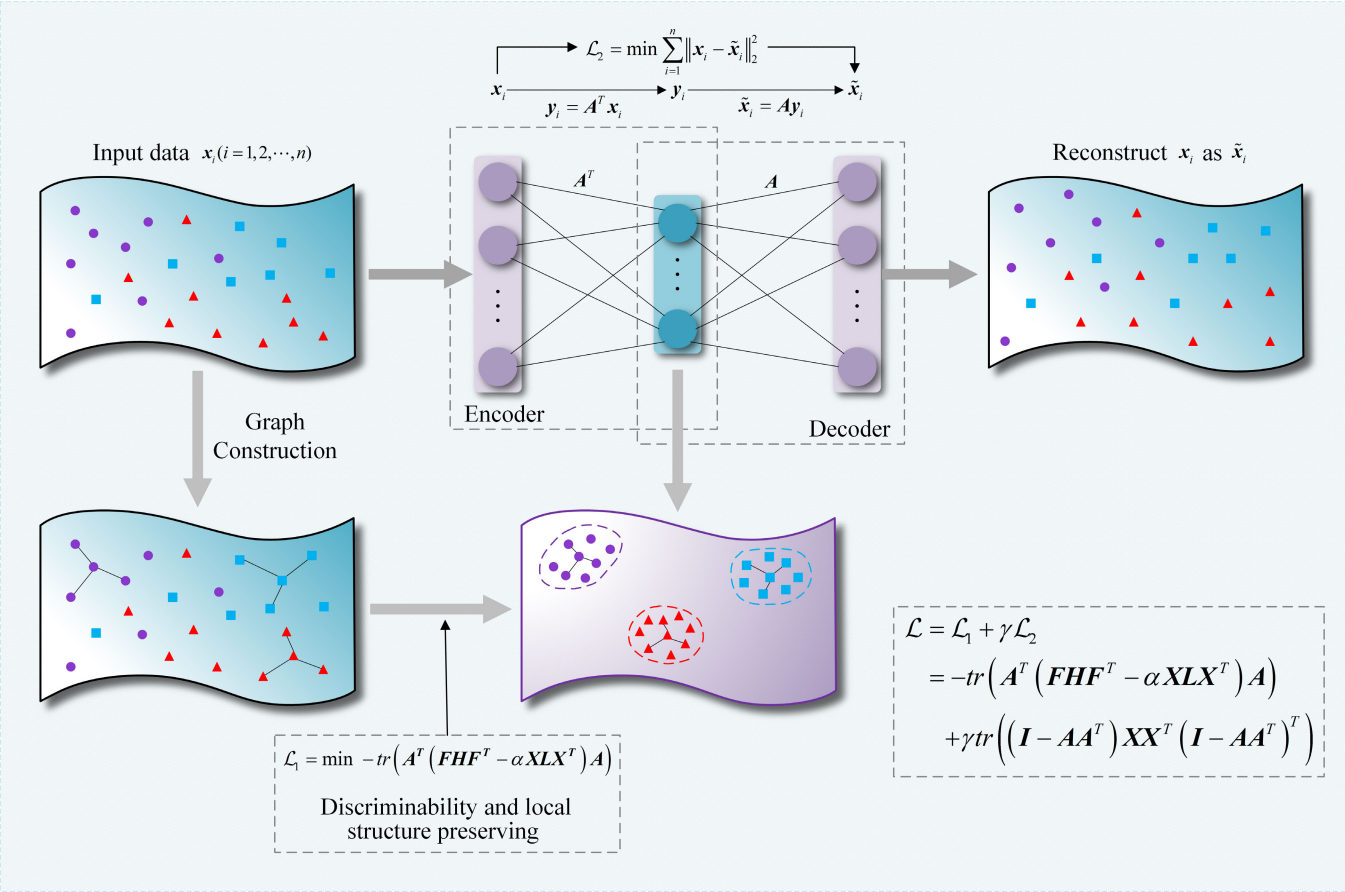

Fig. 1. Framework of the proposed DLPP-AE method.

graph employing the k-nearest neighbor method. The encoding phase involves the DLPP/MMC projection model, where high-dimensional data points $\boldsymbol{x}_i$ are linearly mapped to their low-dimensional counterparts $\boldsymbol{y}_i$ via the linear transformation $\boldsymbol{y}_i = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{x}_i$. Additionally, by virtue of the DLPP/MMC design, this mapping concurrently maximizes interclass scatter while minimizing intraclass scatter, accomplished through the following function:

$$\max \operatorname{tr}\left(\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{S}_b^L - \alpha \boldsymbol{S}_w^L\right)\boldsymbol{A}\right). \tag{3}$$

The DR process represented by (3) in DLPP/MMC can also be expressed as the minimization of (4), serving as the first term of the DLPP-AE objective function, i.e.,

$$\begin{aligned}\mathcal{L}_1 &= -\operatorname{tr}\left(\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{S}_b^L - \alpha \boldsymbol{S}_w^L\right)\boldsymbol{A}\right) \\ &= -\operatorname{tr}\left(\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{F}\boldsymbol{H}\boldsymbol{F}^{\mathrm{T}} - \alpha \boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^{\mathrm{T}}\right)\boldsymbol{A}\right)\end{aligned} \tag{4}$$

where the parameter α is a nonnegative value that balances the relative merits of maximizing the locality-preserving between-class scatter to the minimization of the locality-preserving within-class scatter. In the training process, it is initialized randomly and is then optimized.

The decoding phase involves utilizing a linear autoencoder to reconstruct embedding data back to its original high-dimensional form. Let $\tilde{\boldsymbol{x}}_i$ represent the reconstructed original data point $\boldsymbol{x}_i$, and the weight matrix of the decoder is symbolized by $\boldsymbol{V}^*$. In mathematical terms, the decoding process can be represented by the following formula:

$$\tilde{\boldsymbol{x}}_i = \boldsymbol{V}^* \boldsymbol{y}_i. \tag{5}$$

To streamline the process, it is feasible to set the weight parameters for both the encoding and decoding processes to the same value, following the approach outlined in [25], i.e., $\boldsymbol{V}^* = (\boldsymbol{A}^{\mathrm{T}})^{\mathrm{T}} = \boldsymbol{A}$. Therefore, (5) can be rewritten as

$$\tilde{\boldsymbol{x}}_i = \boldsymbol{V}^* \boldsymbol{y}_i = \boldsymbol{A}\boldsymbol{y}_i = \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{x}_i. \tag{6}$$

Furthermore, our objective also involves minimizing the error between the original input data $\boldsymbol{x}_i$ and its corresponding reconstructed data $\tilde{\boldsymbol{x}}_i$, ensuring that the embedded data more accurately "represent" the original samples, i.e.,

$$\min \sum_{i=1}^{n} \left\| \boldsymbol{x}_i - \tilde{\boldsymbol{x}}_i \right\|_2^2. \tag{7}$$

This process, serving as the second stage of DLPP-AE, is represented as the second term of the DLPP-AE objective function

$$\begin{aligned}\mathcal{L}_2 &= \sum_{i=1}^{n} \left\| \boldsymbol{x}_i - \tilde{\boldsymbol{x}}_i \right\|_2^2 \\ &= \sum_{i=1}^{n} \left\| \boldsymbol{x}_i - \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{x}_i \right\|_2^2 \\ &= \operatorname{tr}\left(\left(\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\right)\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}\left(\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\right)^{\mathrm{T}}\right).\end{aligned} \tag{8}$$

Combining (4), the primary goal of the new methodology is to discover a projection matrix $\boldsymbol{A}$ while simultaneously minimizing the ensuing objective function

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_1 + \gamma\mathcal{L}_2 \\ &= -\mathrm{tr}\left(\boldsymbol{A}^{\mathrm{T}}\left(\boldsymbol{F}\boldsymbol{H}\boldsymbol{F}^{\mathrm{T}} - \alpha\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^{\mathrm{T}}\right)\boldsymbol{A}\right) \\ &\quad + \gamma\mathrm{tr}\left(\left(\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\right)\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}\left(\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\right)^{\mathrm{T}}\right) \end{aligned} \tag{9}$$

where $\alpha(\alpha > 0)$ and γ are the balancing parameters used to denote the relative significance of the associated terms within the objective function.

It is important to highlight that regularization of matrix $\|\boldsymbol{A}\|_F^2$ is not considered in our model. This absence of regularization stems from the fact that the weights of the encoder and decoder are inherently tied together, i.e., $\boldsymbol{A}^* = \boldsymbol{A}$. In other words, regularization of $\|\boldsymbol{A}\|_F^2$ is implicitly addressed through the reconstruction constraint [26].

In (9), the first term $\mathcal{L}_1$ can be considered as the discriminative and locally similar-preserving term, with its coefficient regarded as 1. The second term $\mathcal{L}_2$ can be viewed as the reconstruction term, where the parameter γ denotes the extent of emphasis on reconstruction. When $0 < \gamma < 1$, the proportion of the reconstruction term is relatively smaller compared to that of the discriminative and locally similar-preserving term. If $\gamma > 1$, the proportion assigned to the reconstruction term equals or surpasses that of the discriminative and locally similar-preserving term.

B. Optimization

Different from the linear DR method DLPP, the objective function (9) of the DLPP-AE method is a nonlinear function involving the matrix $\boldsymbol{A}$ and parameters α and γ. While directly solving (9) might be challenging, we can employ gradient descent to find the solution. To enhance the efficiency of the gradient descent learning process, we can utilize the Nesterov accelerated gradient (NAG) [25] method. For DLPP-AE methods involving multiple variables, including the matrix $\boldsymbol{A}$ and parameters α and γ, the multivariate NAG method should be used to optimize parameters.

First, compute the gradient of the objective function (9). The gradient is represented as follows:

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\boldsymbol{A}} &= \frac{d\mathcal{L}_1}{d\boldsymbol{A}} + \gamma\frac{d\mathcal{L}_2}{d\boldsymbol{A}} \\ \frac{\partial\mathcal{L}}{\partial\alpha} &= \mathrm{tr}\left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{A}\right) \\ \frac{\partial\mathcal{L}}{\partial\gamma} &= \mathrm{tr}\left(\left(\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\right)\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}\left(\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\right)^{\mathrm{T}}\right) \end{aligned} \tag{10}$$

where

$$\begin{aligned}\frac{d\mathcal{L}_1}{d\boldsymbol{A}} &= 2\left(\alpha\boldsymbol{X}\boldsymbol{L}\boldsymbol{X}^{\mathrm{T}} - \boldsymbol{F}\boldsymbol{H}\boldsymbol{F}^{\mathrm{T}}\right)\boldsymbol{A} \\ \frac{d\mathcal{L}_2}{d\boldsymbol{A}} &= -4\left(\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\right)\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{A}. \end{aligned} \tag{11}$$

Then, using the gradient from (10), employ the multivariate NAG technique to determine the optimal projection matrix $\boldsymbol{A}$ and parameters α and γ. The process of optimizing parameters using the multivariate NAG method can be summarized as follows: introduce velocity variable, and randomly initialize v, $\boldsymbol{A}$, α, and γ. During the optimization of parameters using traditional gradient descent methods, utilize the velocity variable v to accelerate the optimization process. Taking parameter $\boldsymbol{A}$ as an example, first, update $\boldsymbol{A}$ using the previous velocity v_{t-1}; then, update the velocity variable v_t using the exponentially weighted average gradient; and finally, update the parameter $\boldsymbol{A}_{t+1}$. Iterate these three steps until the optimal parameter $\boldsymbol{A}$ is found. Similarly, the optimal parameters α and γ can be found. The updating process for parameters $\boldsymbol{A}$, α, and γ can be expressed by the following formulas:

$$\begin{aligned}\tilde{\boldsymbol{A}} &= \boldsymbol{A}_t + \rho v_{t-1}, \quad v_t = \rho v_{t-1} - \beta\frac{\partial\mathcal{L}}{\partial\boldsymbol{A}}\left(\tilde{\boldsymbol{A}}, \alpha, \gamma\right) \\ \boldsymbol{A}_{t+1} &= \boldsymbol{A}_t + v_t. \\ \tilde{\alpha} &= \alpha_t + \rho v_{t-1}, \quad v_t = \rho v_{t-1} - \beta\frac{\partial\mathcal{L}}{\partial\alpha}\left(\boldsymbol{A}_t, \tilde{\alpha}, \gamma\right) \\ \alpha_{t+1} &= \alpha_t + v_t. \\ \tilde{\gamma} &= \gamma_t + \rho v_{t-1}, \quad v_t = \rho v_{t-1} - \beta\frac{\partial\mathcal{L}}{\partial\gamma}\left(\boldsymbol{A}_t, \alpha, \tilde{\gamma}\right) \\ \gamma_{t+1} &= \gamma_t + v_t. \end{aligned} \tag{12}$$

Algorithm 1 Optimize the Parameters $\boldsymbol{A}$, α, and γ

Require: the input data matrix $\boldsymbol{X}$,the between-class weight matrix $\boldsymbol{B}$ and within-class weight matrix $\boldsymbol{W}$.

Ensure: the optimal projection matrix $\boldsymbol{A}$, the parameter α and γ.

1: Initialize velocity variable v, the parameters $\boldsymbol{A}$, α and γ randomly, set a learning rate ρ of gradient descent, and set a threshold ϵ for the change of loss function.
2: Calculate the between-class Laplacian matrix $\boldsymbol{H}$, the within -class Laplacian matrix $\boldsymbol{L}$, and the objective function $\mathcal{L}$ using Eq.(9).
3: **while** the change in the loss function $> \epsilon$ **do**
4: Calculate $\partial\mathcal{L}/\partial\boldsymbol{A}$, $\partial\mathcal{L}/\partial\alpha$, $\partial\mathcal{L}/\partial\gamma$ using Eq.(10)
5: Update $\boldsymbol{A}$, α and γ using Eq.(12)
6: Calculate using Eq.(9)
7: **end while**

The optimization process for parameters $\boldsymbol{A}$, α, and γ can be summarized in Algorithm 1. First, construct weight matrices $\boldsymbol{W}$ and $\boldsymbol{B}$ to represent within-class and between-class relationships, respectively. Then, Laplacian matrices $\boldsymbol{L}$ and $\boldsymbol{H}$ are computed based on these weight matrices. Next, it is necessary to initialize some variables randomly, including velocity variable v and parameters $\boldsymbol{A}$, α, and γ. Specifically, the learning rate β is set to $5 * 10^{-3}$, the tolerance threshold for the change in the loss function ϵ is set at 0.05, and the momentum coefficient ρ is set to 0.6. Following that, we begin the iterative process of Algorithm 1. Upon completion of the iterations, the optimal projection matrix $\boldsymbol{A}$ is obtained. Through these steps, we accomplish the training process of the DLPP-AE method.

C. Application of DLPP-AE in Fault Diagnosis

Adhering to Algorithm 1, the DLPP-AE model can undergo training and be applied within fault diagnosis applications. The

workflow of the fault diagnosis model leveraging DLPP-AE is depicted in Fig. 2.

First, for a given training set of samples $M \times N$ matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N]$, an adjacency graph is constructed based on k-nearest neighbors. Then, the DLPP-AE algorithm is employed for feature extraction to compute the DR matrix $\boldsymbol{A}$. Once the DR matrix is acquired, it becomes imperative to reduce the dimensionality of the training data to a suitable extent. In the field of DR, the AIC is commonly used to determine the optimal number of dimensions. The AIC assesses the complexity of statistical models and measures the goodness of fit of the statistical model to the data. Rooted in the notion of entropy, the AIC seeks to strike a balance between the complexity of the estimated model and its efficacy in fitting the data accurately [27]. Next, the training set $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]$ and the test set $\boldsymbol{X}' = [\boldsymbol{x}_1', \boldsymbol{x}_2', \ldots, \boldsymbol{x}_t']$ are projected into a low-dimensional space of dimensionality d using the DR matrix $\boldsymbol{A}$, resulting in low-dimensional embedding data $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n]$ and $\boldsymbol{Y}' = [\boldsymbol{y}_1', \boldsymbol{y}_2', \ldots, \boldsymbol{y}_t']$, respectively. Finally, the fault training set's projected data $\boldsymbol{Y}$ are used to train a classifier, and then, test the classifier using the projected data $\boldsymbol{Y}'$ from the fault test set to calculate the fault detection accuracy.

IV. Case Study

In this section, experiments are conducted on four datasets to validate the effectiveness of the proposed DLPP-AE-based fault diagnosis method. These datasets include a chemical process fault dataset, two mechanical fault datasets, and one real-world industrial fault dataset. To comprehensively evaluate the performance of the proposed DLPP-AE method, we report not only classification accuracy but also precision, recall, and F1-score as additional metrics. It is important to note that for this multiclass classification problem, the reported precision, recall, and F1-scores are macro-averaged, meaning that these metrics are calculated for each class and then averaged. All results presented are based on threefold cross-validation repeated three times. Specifically, one part of the redivided dataset was used for training, while the other two parts were used for testing.

Additionally, we evaluated the performance of the proposed approach by comparing it with several traditional methods, including PCA [13], LPP, NPE [28], DLPP, MMC [29], DLPP/MMC, MC-DLPP [30], A-DLPP-R [20], and RFNDNLP-NB [31]. The AIC was employed to determine the optimal dimensionality for DR. After applying these methods for DR on the extracted features, a KNN classifier was used for feature classification and fault identification. Furthermore, to validate the effectiveness of DLPP-AE, we compared its performance with several state-of-the-art deep neural network models from recent years, such as ICNN [32], MS-CNN [33], and EVGG [34]. These models were also evaluated using the KNN classifier. Please note that due to the fixed network structures of the three deep neural network methods, they cannot process low-dimensional data with only 52 features, such as the TE dataset. Therefore, these three deep learning methods were only compared on three other datasets. Finally, to gain a deeper understanding of the relationships between fault variables and enhance the credibility of the experimental analysis, we also conducted an analysis of the model's interpretability.

The algorithm with MATLAB R2023a runs on a PC with Intel[1] Core[2] i7-7500U CPU at 2.70 GHz, 8-GB memory under the Windows 11 operation system.

A. Experimental Datasets

The TE process is a simulation platform designed by Eastman Chemical Company, capable of simulating 21 common faults and disturbances encountered in real chemical industry operations [35]. The TE process includes 41 measurement variables and 11 control variables, resulting in a dimensionality of 52 for TE data. The entire TE dataset consists of 22 class labels, including one class of normal data and 21 different fault types. Each data class comprises both a training set and a test set. The training set for normal data encompasses 500 samples, while the test set comprises 960 samples. For fault data, each training set comprises 480 samples, and the corresponding test set contains 960 samples. The complete TE process diagram is depicted in Fig. 3.

The first bearing dataset (CWRU) is provided by Case Western Reserve University [36]. The CWRU bearing dataset was collected using an experimental platform consisting of a 1.5-kW motor, drive-end bearing, fan-end bearing, torque sensor, dynamometer, accelerometers, and an electronic controller. Vibration signals were acquired using a 16-channel data recorder at a sampling frequency of 12 kHz, while power and rotational speed were measured via a torque sensor. The raw data include four different bearing health conditions: normal, inner race fault, outer race fault, and rolling element fault, each with fault diameters of 7, 14, and 21 mil. In the experiments, the data were resampled to a size of 1×128.

The second bearing dataset (Polito) was collected using an experimental platform at the Department of Mechanical and Aerospace Engineering, Politecnico di Torino, specifically designed for testing high-speed aerospace bearings [37]. The data were gathered under variable rotational speeds, radial loads, and damage levels. The dataset includes three health conditions: inner race fault, rolling element fault, and healthy bearings. For both the inner race fault and the rolling element fault, data were collected for three different fault sizes: 150, 250, and 450 μm. Data for each bearing type were collected at frequencies of 100, 200, 300, 400, and 500 Hz. During the experiment, the data were resampled to a size of 1×128.

The real industrial machinery fault dataset (SCA) comes from Mid Sweden University and Svenska Cellulosa Aktiebolaget [38]. The main difference from general mechanical fault datasets is that the SCA dataset was collected in a real industrial environment, rather than being simulated in a laboratory. This dataset provides bearing fault data under ten different conditions, including ball faults, inner faults, outer faults, and one case of external event impact on the vibration signal, as detailed in [38]. In our experiment, we selected five

[1]Registered trademark.
[2]Trademarked.

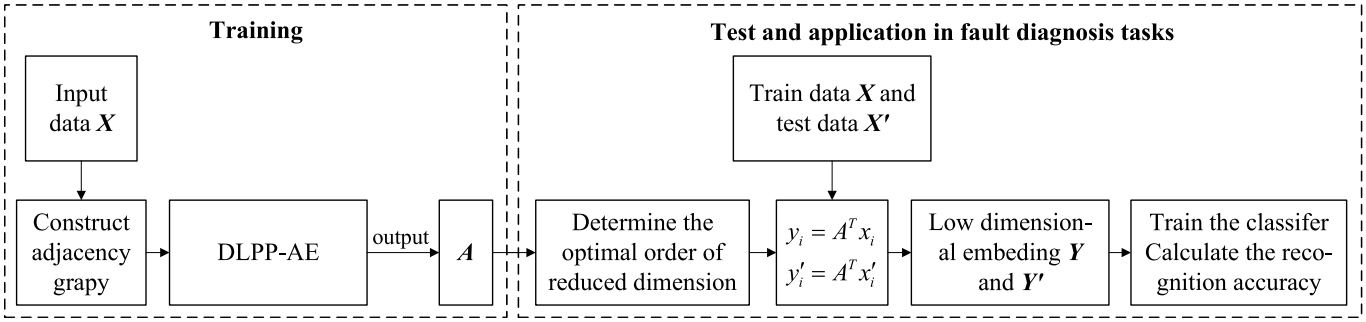

Fig. 2. Workflow based on DLPP-AE fault diagnosis model.

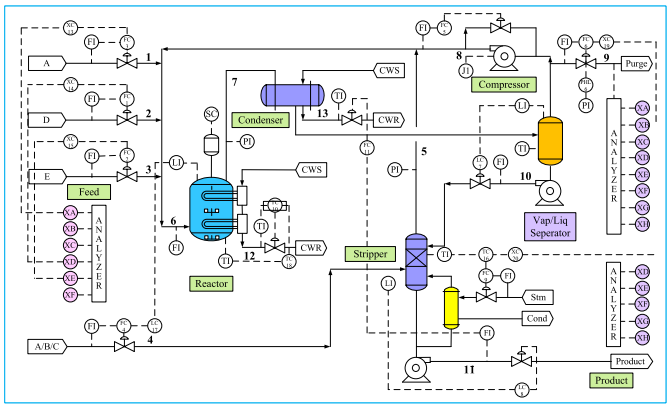

Fig. 3. Flowchart of the TE chemical process.

different types of fault data and resampled them to a size of 1×128.

The details of the datasets are listed in Table I.

B. Experiment on Dataset TE

The TE dataset contains data on 21 common process faults observed in chemical industrial operations. To validate the effectiveness of DLPP-AE in fault diagnosis, we utilized the cases presented in [30], specifically faults 1, 4, 5, 7, and 10. For each fault class, 400 training samples and 800 test samples are extracted for simulation experiments.

First, Table II lists the classification accuracies of various algorithms on the TE dataset. To better evaluate the performance of the DLPP-AE method, Table II also includes precision, recall, and F1-scores.

As shown in Table II, DLPP-AE outperforms the PCA, LPP, NPE, DLPP, MMC, DLPP/MMC, MC-DLPP, A-DLPP-R, and RFNDNLP-NB methods across all evaluation metrics. The classification accuracy of DLPP-AE has reached 99.58%, showing an improvement of approximately 14.02% compared to the MC-DLPP method proposed in 2021 and 9.08% compared to the RFNDNLP-NB method proposed in 2022. This indicates that DLPP-AE effectively enhances the diagnostic capability of feature extraction. In terms of precision, DLPP-AE achieved an outstanding 99.5%, which is nearly 5.11% higher than the second-best algorithm, DLPP/MMC. When evaluating recall, DLPP-AE again outperformed, attaining 99.71%, which reflects its high sensitivity and effectiveness in capturing all actual fault instances. This recall rate surpasses DLPP by 8.34% and achieves an increase of 13.15% over the MC-DLPP method. The DLPP-AE demonstrated reliability in fault detection without missing true positives. The F1-score of DLPP-AE further illustrates its balanced capability between precision and recall, achieving a score of 99.6%. Table II signifies a 7.56% enhancement over DLPP and an impressive 13.03% increase compared to MC-DLPP, indicating the model's balanced accuracy and recall in fault diagnostics. Overall, these results indicate that DLPP-AE not only improves diagnostic accuracy, but also establishes a new benchmark for robust feature extraction in fault diagnosis, consistently outperforming alternative methods across multiple metrics.

In order to offer a more lucid and comprehensive visualization of the experimental findings from an intuitive standpoint, Fig. 4 also presents a colormap. The x-axis of the color mapping chart denotes the quantity of samples, while the y-axis corresponds to the types of samples. The color mapping visually displays the number and location of misclassifications. The greater the degree of color blending, the higher the misclassification rate, whereas, purer colors indicate higher classification accuracy. From Fig. 4, it can be observed that the color mapping for fault diagnosis based on DLPP-AE is purer compared to PCA, LPP, NPE, DLPP, MMC, and DLPP/MMC, indicating higher accuracy in fault diagnosis based on DLPP-AE.

Finally, to further explore the classification performance, Fig. 5 presents the 3-D visualization of features extracted after DR by different methods. In these visualizations, the proposed DLPP-AE achieves the best separation of fault features, with slight overlap between fault 5 and fault 10. The clustering effect of fault features extracted by DLPP is relatively lower. For other models, nearly all fault features are intermingled, and no fault can be separated clearly by 100%. Among these models, the proposed DLPP-AE achieves the best clustering performance.

C. Experiment on Dataset CWRU

The CWRU bearing dataset is commonly used to evaluate the performance of algorithms. It includes data under four conditions: normal, inner race fault, outer race fault, and ball fault. The inner race, outer race, and ball faults have three different fault diameters: 7, 14, and 21 mils. Thus, the CWRU dataset is subdivided into ten fault types. To validate the effectiveness of DLPP-AE in fault diagnosis, we selected five fault types: outer race (7 mil), inner race (14 mil), outer race

TABLE I
DETAILS OF THE DATASETS

Datasets	Class Number	Sample Number/class	Sample Size	Environment
TE(Eastman Chemical Company)	5	400	1 × 52	Laboratory
CWRU(Case Western Reserve University)	5	312	1 × 128	Laboratory
Polito(Politecnico di Torino)	4	312	1 × 128	Laboratory
SCA(Mid Sweden University)	5	682	1 × 128	Industry

TABLE II
COMPARISON OF CLASSIFICATION PERFORMANCE (%) OF DIFFERENT METHODS ON DATASET TE

Method	Accuracy	Precision	Recall	F1
LPP	86.23	89.63	86.21	87.82
DLPP	90.75	92.73	91.37	92.04
MMC	47.98	40.49	41.96	40.74
PCA	83.95	83.76	84.08	83.91
NPE	92.18	93.89	89.92	93.04
DLPP/MMC	93.65	94.39	95.87	95.12
A-DLPP-R[20]	87.74	86.02	87.83	86.24
MC-DLPP[30]	85.56	90.22	85.56	86.57
RFNDNLP-NB[31]	90.5	89.06	90.49	89.71
DLPP-AE	**99.58**	**99.5**	**99.71**	**99.6**

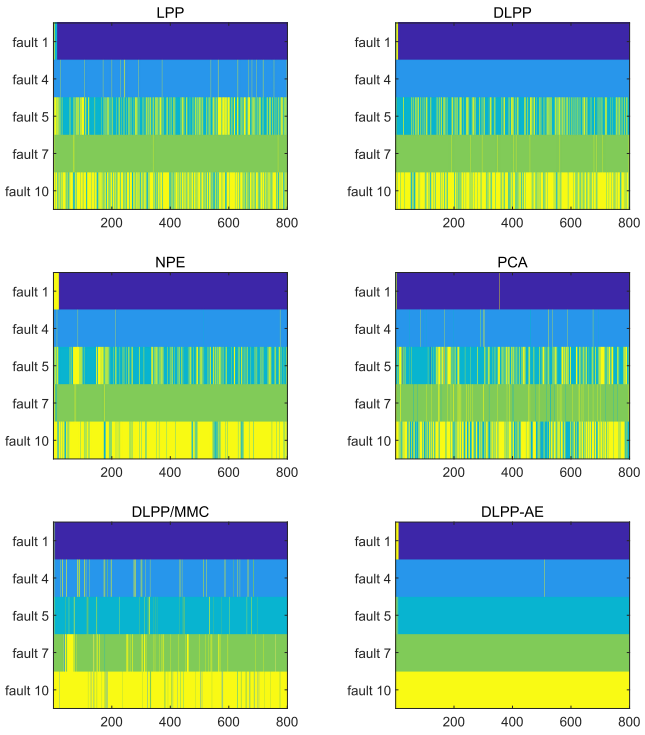

Fig. 4. Colormap on dataset TE.

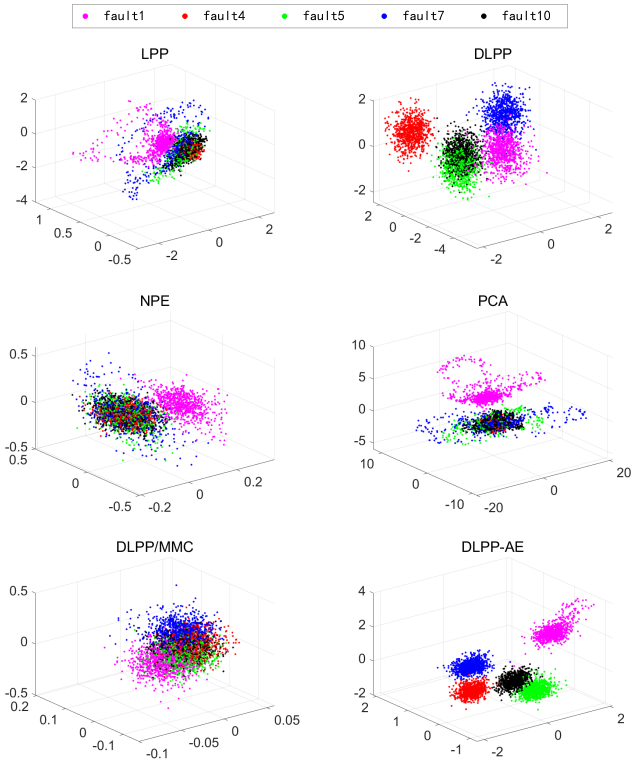

Fig. 5. Feature visualization on dataset TE.

TABLE III
COMPARISON OF CLASSIFICATION PERFORMANCE (%) OF DIFFERENT METHODS ON DATASET CWRU

Method	Accuracy	Precision	Recall	F1
LPP	72.31	69.11	68.51	68.65
DLPP	98.77	98.88	98.26	98.57
MMC	34.78	43.84	35.60	38.39
PCA	95.25	93.71	95.73	94.63
NPE	90.38	88.34	87.34	86.7
DLPP/MMC	93.01	93.86	95.81	94.79
ICNN	91.36	90.78	91.93	91.35
MS-CNN	95.86	96.77	95.83	96.29
EVGG	93.62	93.97	93.86	93.91
DLPP-AE	**100**	**100**	**100**	**100**

(14 mil), ball (14 mil), and inner race (21 mil), labeled as fault 3, fault 5, fault 6, fault 7, and fault 8, respectively. For each fault type, 312 training samples and 624 testing samples were extracted for simulation experiments. For the deep neural network methods, including ICNN, MS-CNN, and EVGG, the number of epochs was set to 100.

According to Table III, DLPP-AE achieved outstanding results across all evaluation metrics on the CWRU dataset. Specifically, for fault classification accuracy, DLPP-AE achieved an impressive 100%, significantly outperforming traditional methods such as LPP (72.31%) and DLPP/MMC (93.01%), as well as more recent deep neural network methods such as ICNN (91.36%) and MS-CNN (95.86%). Regarding precision, DLPP-AE also reached 100%, surpassing the second-best traditional method, DLPP (98.88%), and the top deep neural network method, MS-CNN (96.77%). For recall and F1-score, DLPP-AE also outperformed all other comparison methods. Overall, these results demonstrate that the proposed DLPP-AE not only improves diagnostic accuracy on the CWRU dataset, but also proves its excellent capability in reliably detecting faults without missing true positives, as well as balancing accuracy and recall effectively.

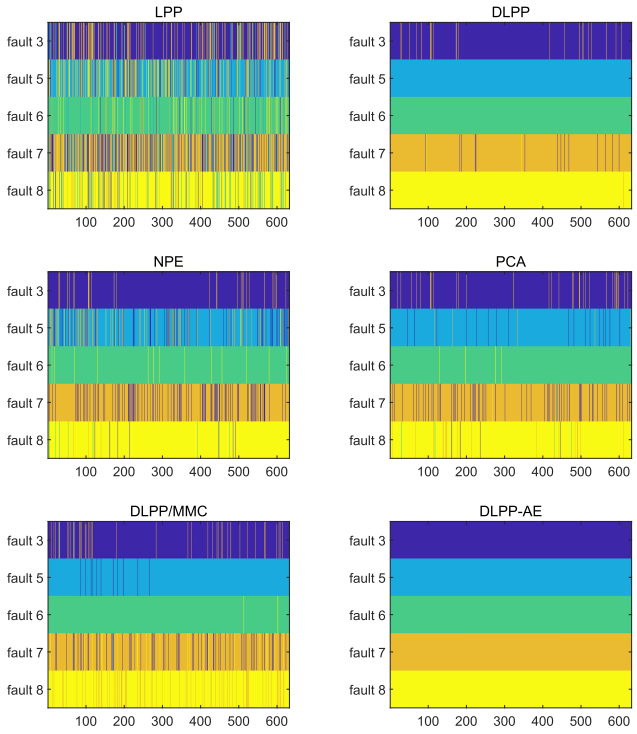

Fig. 6. Colormap on dataset CWRU.

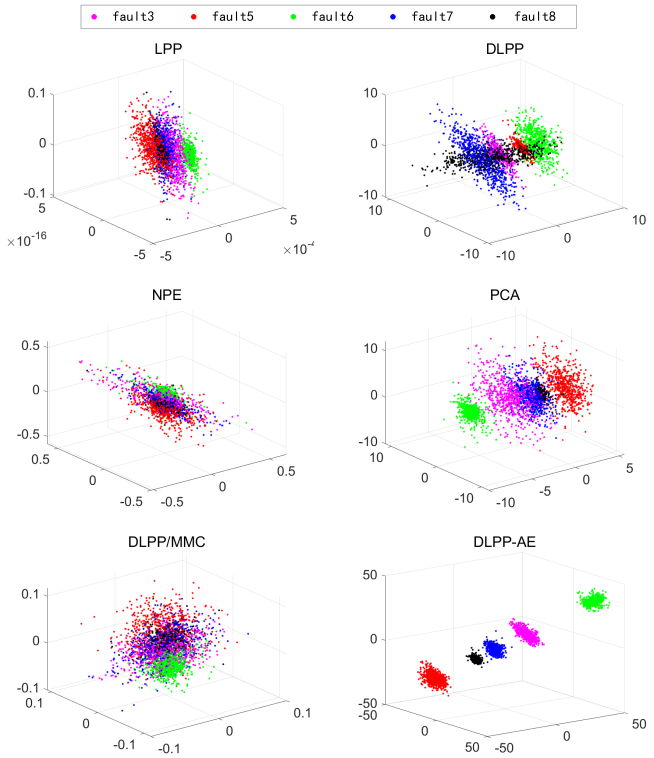

Fig. 7. Feature visualization on dataset CWRU.

From colormaps in Fig. 6, it can be observed that DLPP-AE achieved a 100% classification accuracy. However, all other methods misclassified some faults. For instance, both DLPP/MMC and NPE tend to misclassify fault 7 as fault 3. In terms of fault types, fault 6 is relatively easier to identify, as almost all methods, except for LPP, were able to correctly classify fault 6.

Fig. 7 displays the 3-D visualization of the features extracted after DR using different methods. Among these visualizations, the proposed DLPP-AE achieves the best separation of fault features. The fault feature clustering for PCA is relatively weak. For other models, almost all the fault features are mixed, with no fault being 100% clearly separated. Among these models, the proposed DLPP-AE demonstrates the best clustering performance.

D. Experiment on Dataset Polito

The Polito dataset includes data under three conditions: normal, inner race fault, and ball fault. Data were collected at 100, 200, 300, 400, and 500 Hz for each fault type. For DLPP-AE validation, we selected inner race fault and rolling element fault data at 100 Hz, as well as inner race fault and rolling element fault data at 300 Hz, labeled as fault 1, fault 2, fault 3, and fault 5, respectively. For each fault type, 312 training samples and 624 testing samples were extracted for simulation experiments. For the deep neural network methods, including ICNN, MS-CNN, and EVGG, the number of epochs was set to 100.

As shown in Table IV, DLPP-AE demonstrates superior performance over other algorithms on the Polito dataset. The fault classification accuracy of DLPP-AE reached 99.09%, surpassing traditional methods such as DLPP (96.48%) and LPP (91.26%), as well as deep learning methods such as ICNN (97.66%) and MS-CNN (81.25%). For precision, recall, and F1-score, the DLPP-AE algorithm achieved 98.81%, 98.79%, and 98.79%, respectively, demonstrating comprehensive superiority over other comparison methods.

TABLE IV
COMPARISON OF CLASSIFICATION PERFORMANCE (%) OF DIFFERENT METHODS ON DATASET POLITO

Method	Accuracy	Precision	Recall	F1
LPP	91.26	90.20	88.50	88.80
DLPP	96.48	95.88	95.31	95.49
MMC	34.18	34.04	33.7	33.75
PCA	88.21	87.87	84.49	85.03
NPE	52.85	80.08	37.13	26.15
DLPP/MMC	96.72	95.84	95.62	95.66
ICNN	97.66	97.73	97.66	97.64
MS-CNN	81.52	89.35	81.52	78.62
EVGG	98.28	98.30	98.28	98.28
DLPP-AE	**99.09**	**98.81**	**98.79**	**98.79**

Fig. 8 illustrates the details of fault classification for different methods. It can be observed that DLPP-AE achieves nearly perfect fault classification, while other methods exhibit misclassification for certain fault types. For example, PCA tends to misclassify fault 3 as fault 2, and DLPP/MMC shows lower classification accuracy for fault 1. In all fault types, DLPP-AE maintained stable and excellent performance. Overall, DLPP-AE not only excels in classification accuracy, precision, recall, and F1-score, but also proves its excellent capability in fault feature extraction and classification.

Fig. 9 further illustrates the 3-D visualization of the features extracted after DR using different methods. Among these visualizations, DLPP-AE demonstrates the best feature

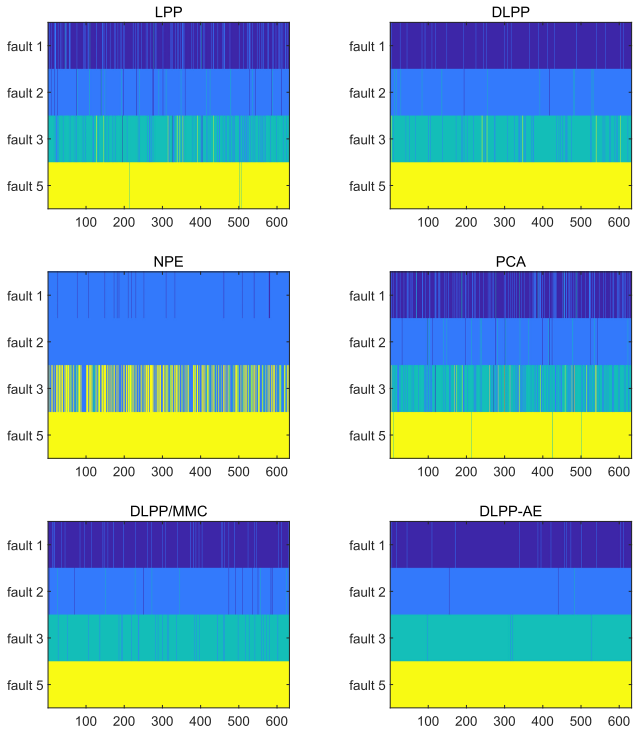

Fig. 8. Colormap on dataset Polito.

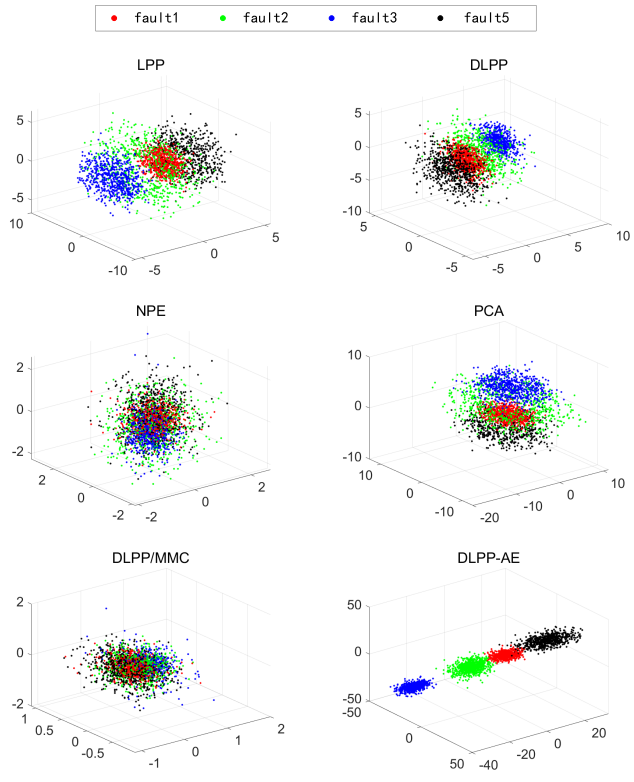

Fig. 9. Feature visualization on dataset Polito.

separation capability. In contrast, traditional methods such as PCA and DLPP/MMC only achieve limited clustering, with extracted features still showing noticeable overlap. DLPP-AE successfully separates the features of each fault type, with no significant mixing.

TABLE V
COMPARISON OF CLASSIFICATION PERFORMANCE (%) OF DIFFERENT METHODS ON DATASET SCA

Method	Accuracy	Precision	Recall	F1
LPP	77.77	54.81	65.32	54.66
DLPP	58.07	45.56	57.64	50.89
MMC	85.90	83.47	87.16	85.15
PCA	63.41	74.72	59.74	57.86
NPE	77.25	77.21	70.38	69.26
DLPP/MMC	57.47	59.87	64.03	59.14
ICNN	81.71	82.25	81.68	81.96
MS-CNN	88.00	88.97	87.28	88.11
EVGG	82.39	81.64	81.45	81.54
DLPP-AE	**92.04**	**91.35**	**90.21**	**90.58**

E. Experiment on Dataset SCA

For the SCA dataset, we selected five fault types for validation: outer race fault 1, inner race fault 1, ball fault, outer race fault 2, and inner race fault 2, which are named as fault 1, fault 3, fault 5, fault 9, and fault 10. For each sample type, 682 training samples and 1364 testing samples were extracted for the experiments. Since the SCA dataset is collected from real industrial machinery, it is highly suitable for testing the fault diagnosis capabilities of algorithms under real, complex industrial conditions. The number of epochs for deep learning methods is set to 100. The evaluation results are listed in Table V.

From Table V, it can be concluded that, compared to standard laboratory datasets, the fault diagnosis performance of various methods in real industrial scenarios has decreased. However, the performance of DLPP-AE remains consistent with that observed in standard laboratory datasets, meaning that it still significantly outperforms other methods. Specifically, DLPP-AE achieves a fault classification accuracy of 92.04%, which is an impressive 34.57% higher than DLPP/MMC, 10.3% higher than ICNN proposed in 2022, and 10.04% higher than EVGG proposed in 2023. The precision, recall, and F1-score of DLPP-AE are 91.35%, 90.21%, and 90.58%, respectively. This indicates that DLPP-AE maintains a significant advantage in real industrial scenarios, not only leading in classification accuracy but also performing excellently in other key metrics. In contrast, DLPP/MMC shows a large performance drop in real industrial data, while DLPP-AE only experiences a slight decrease in classification accuracy, demonstrating its greater robustness and adaptability to complex and noisy real industrial data.

To further validate the algorithm's performance on real industrial datasets, Fig. 10 presents the colormap, and Fig. 11 shows the 3-D visualization of features extracted with different methods. From Fig. 10, it can be observed that fault 1 and fault 3 share significant similarities and are often misclassified. Compared to the many cluttered colormaps from other methods, DLPP-AE maintains a higher level of purity in its colormap. Fig. 11 supports these findings, showing that, except for DLPP-AE, the features from other methods are heavily mixed. DLPP-AE shows some overlap between fault 1 and fault 3, but all other features are well-separated, demonstrating the best clustering performance.

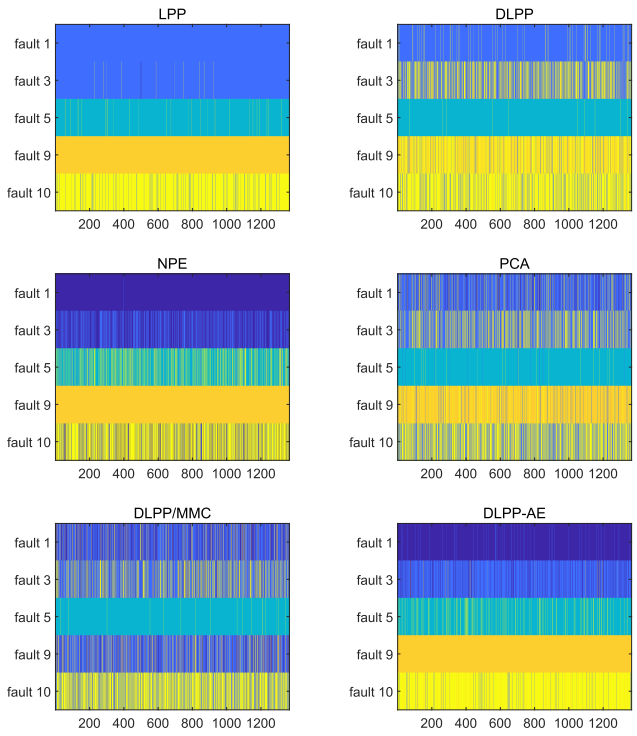

Fig. 10. Colormap on dataset SCA.

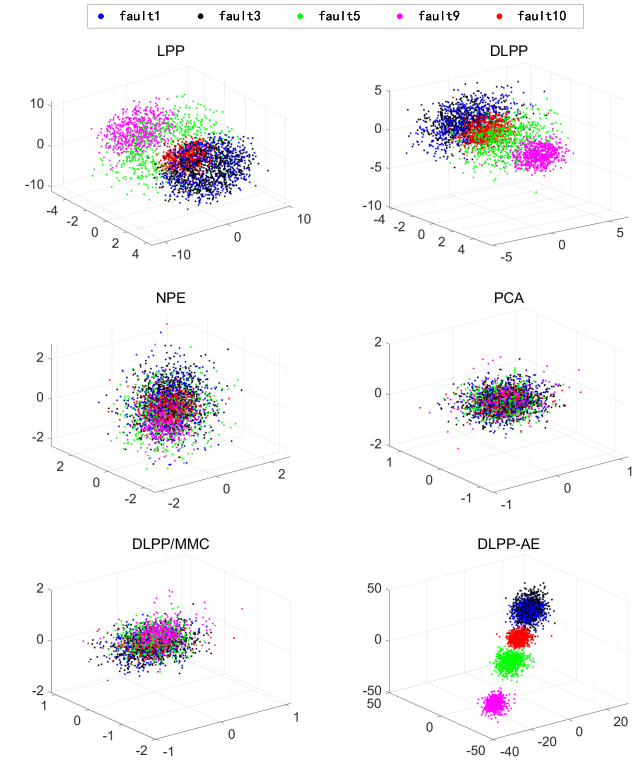

Fig. 11. Feature visualization on dataset SCA.

Overall, the validation of the SCA dataset further confirms the good performance of DLPP-AE. Not only does it perform well in laboratory environments, but it also exhibits strong generalization capability in real industrial scenarios. This outstanding performance is attributed to the design of DLPP-AE in feature extraction and fault pattern recognition, enabling it to achieve efficient and accurate diagnostics even in complex environments.

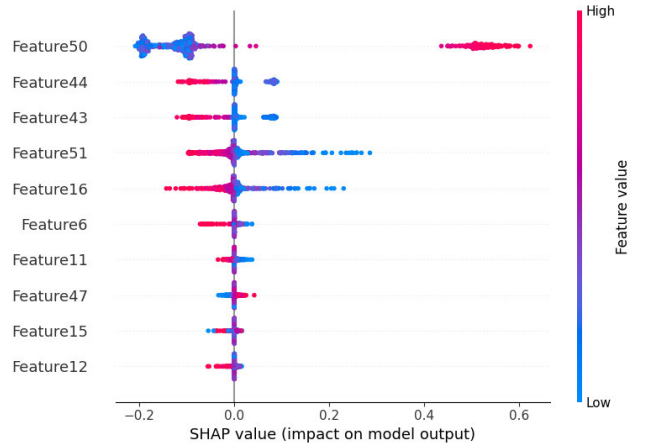

Fig. 12. Feature importance ranking (top 10) and impact analysis of each feature for fault 1.

F. Interpretability

For the interpretability, the SHapley Additive exPlanations (SHAP) method [39] is used to explain model predictions, using fault 1 from the TE dataset as an example. SHAP is a method for explaining the predictions of machine learning models. It explains the model's output by assigning importance values to features, with its core principle based on the concept of Shapley Value from game theory. The SHAP framework can be applied to explain various complex models. Generally, SHAP values are calculated based on both global and individual samples to provide model interpretability.

To analyze the global interpretability of the proposed model, we generated a SHAP value summary figure, as shown in Fig. 12. In this figure, the top ten most important features of fault 1 and their respective impacts are visualized, where the horizontal axis represents the SHAP value, indicating the impact of a feature on the model's predictions, and the vertical axis represents the top ten features in terms of importance ranking. The SHAP values of different features are distributed on both sides of the baseline in the middle. The left area represents the negative impact of influencing factors on the results, while the right area represents the positive impact of influencing factors on the results. Each sample in the dataset is run through the SHAP explainer, and a point result is created for each feature. The color of each point varies according to the size of its feature value. This visual representation provides an intuitive understanding of how each feature affects prediction and its importance.

In Fig. 12, Feature 50 (the flow rate of the reactor water cooling system) is made an example. As the flow rate of reactor water increases, it makes a positive contribution to the prediction results of fault 1, that is, the higher the flow rate of reactor water, the higher the probability of fault 1 occurrence. As the flow rate of reactor water decreases, it has a negative impact on the prediction results of fault 1, that is, the smaller the value, the lower the probability of fault 1 occurrence. In the process of chemical production, decreasing the flow rate of

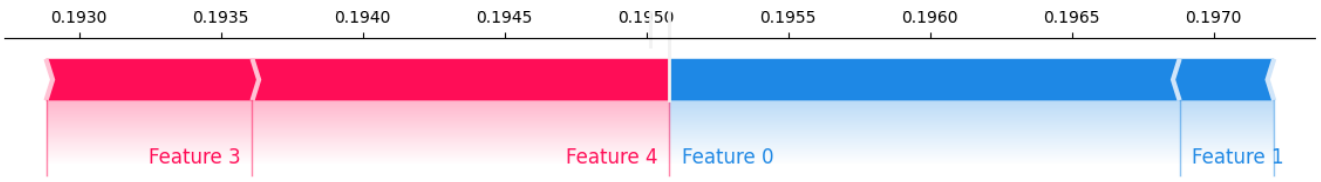

Fig. 13. Interpretability of individual sample for fault 1.

reactor water cooling system appropriately can help maintain the stable operation of the equipment.

To study the interpretability at the sample level, we analyzed an individual sample classified as fault 1. The results are presented in Fig. 13, where a sample predicted as fault 1 is randomly selected for analysis, and the contributions of specific features to the final prediction are visualized. This figure shows the contribution made by each feature in the sample in pushing the model's prediction results from the average predicted value (base value) to the final predicted value. The red-marked area reflects the feature with a positive SHAP value, and the area size of this area intuitively displays the absolute amount of the corresponding SHAP value, which also means that this feature has a more significant contribution to the model's decision results. On the contrary, the blue marked area represents features with negative SHAP values, which serve to reduce the model's predicted values.

In Fig. 13, features with positive SHAP values, such as Feature 3 and Feature 4, are highlighted in red, indicating their positive contribution to the fault classification. Conversely, features such as Feature 0 and Feature 1, with negative SHAP values, are highlighted in blue, showing their negative impact on the prediction. Features such as Feature 2 exhibit relatively small SHAP values, indicating minimal influence on the prediction outcome. This analysis demonstrates how individual features interact to push the model's prediction from the base value to the final output.

Types of faults can be interpreted similarly.

G. Analysis

From this study, the following conclusions can be drawn.

1) The fault diagnosis performance of DLPP-AE surpasses that of other algorithms (such as the original DLPP/MMC, traditional DLPP, PCA, MMC, MC-DLPP, RFNDLP-NB, and deep neural networks such as ICNN, MS-CNN, and EVGG). For instance, on the TE dataset, DLPP-AE's fault classification accuracy is 5.39%, 8.83%, 15.63%, and 9.08% higher than that of DLPP/MMC, DLPP, PCA, and RFNDLP-NB, respectively.
2) DLPP-AE demonstrates better generalization capabilities compared to other algorithms, as evidenced by experiments on the CWRU and Polito datasets. Specifically, compared to other algorithms, DLPP-AE's accuracy on the CWRU dataset is 1.23%–65.22% higher.
3) The DLPP-AE algorithm is suitable for feature extraction from various industrial data. For example, it has been used in chemical industry process fault datasets and mechanical fault datasets in both standard laboratory and real-world industrial environments to prove the effectiveness of the DLPP-AE fault diagnosis method.

V. Conclusion

This article introduces a novel DLPP method referred to as DLPP-AE. DLPP-AE utilizes the reconstruction mechanism of autoencoders to establish a bidirectional mapping, which enables data points in the latent space to be reconstructed back to their original form. This effectively resolves the issue of DLPP having only a unidirectional mapping, resulting in less accurate low-dimensional features. We conducted fault diagnosis experiments on a chemical industrial process fault dataset, two mechanical fault datasets, and one real-world industrial fault dataset, providing a comprehensive evaluation of the proposed DLPP-AE method. In these experiments, DLPP-AE was compared with various fault diagnosis methods, including traditional approaches such as PCA, LPP, NPE, DLPP, MMC, DLPP/MMC, MC-DLPP, A-DLPP-R, and RFNDNLP-NB, as well as the state-of-the-art deep neural network methods such as ICNN, MS-CNN, and EVGG. The experimental results demonstrated that the proposed DLPP-AE method exhibits superior fault feature extraction and generalization capabilities. Future work will focus on exploring the application of more advanced autoencoder techniques, such as stacked sparse autoencoders, to further enhance diagnostic performance.

References

[1] Z. Yang, B. Xu, W. Luo, and F. Chen, "Autoencoder-based representation learning and its application in intelligent fault diagnosis: A review," *Measurement*, vol. 189, Feb. 2022, Art. no. 110460.
[2] Z. Zhu et al., "A review of the application of deep learning in intelligent fault diagnosis of rotating machinery," *Measurement*, vol. 206, Jan. 2023, Art. no. 112346.
[3] M. Rezamand, M. Kordestani, R. Carriveau, D. S.-K. Ting, M. E. Orchard, and M. Saif, "Critical wind turbine components prognostics: A comprehensive review," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 12, pp. 9306–9328, Dec. 2020.
[4] W. Deng, Z. Li, X. Li, H. Chen, and H. Zhao, "Compound fault diagnosis using optimized MCKD and sparse representation for rolling bearings," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–9, 2022.
[5] Y. Li, X. Zhang, Z. Chen, Y. Yang, C. Geng, and M. J. Zuo, "Time-frequency ridge estimation: An effective tool for gear and bearing fault diagnosis at time-varying speeds," *Mech. Syst. Signal Process.*, vol. 189, Apr. 2023, Art. no. 110108.
[6] N. Md Nor, C. R. Che Hassan, and M. A. Hussain, "A review of data-driven fault detection and diagnosis methods: Applications in chemical process systems," *Rev. Chem. Eng.*, vol. 36, no. 4, pp. 513–553, May 2020.
[7] A. S. Maliuk, A. E. Prosvirin, Z. Ahmad, C. H. Kim, and J.-M. Kim, "Novel bearing fault diagnosis using Gaussian mixture model-based fault band selection," *Sensors*, vol. 21, no. 19, p. 6579, Oct. 2021.
[8] Z. Chen, Q. Zhang, J. Zhang, X. Qin, and Y. Sun, "A knowledge-based fault diagnosis method for rolling bearings without fault sample training," *Proc. Inst. Mech. Eng., C, J. Mech. Eng. Sci.*, vol. 238, no. 20, pp. 10253–10265, Oct. 2024.

[9] D. Zhao et al., "Enhanced data-driven fault diagnosis for machines with small and unbalanced data based on variational auto-encoder," *Meas. Sci. Technol.*, vol. 31, no. 3, Mar. 2020, Art. no. 035004.

[10] H. Zhao, H. Liu, Y. Jin, X. Dang, and W. Deng, "Feature extraction for data-driven remaining useful life prediction of rolling bearings," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–10, 2021.

[11] J. Cen, Z. Yang, X. Liu, J. Xiong, and H. Chen, "A review of data-driven machinery fault diagnosis using machine learning algorithms," *J. Vib. Eng. Technol.*, vol. 10, pp. 2481–2507, Oct. 2022.

[12] J. Zhang, J. Yu, and D. Tao, "Local deep-feature alignment for unsupervised dimension reduction," *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2420–2432, May 2018.

[13] Y. Han, J. Liu, F. Liu, and Z. Geng, "An intelligent moving window sparse principal component analysis-based case based reasoning for fault diagnosis: Case of the drilling process," *ISA Trans.*, vol. 128, pp. 242–254, Sep. 2022.

[14] Z. Du and X. Jin, "Multiple faults diagnosis for sensors in air handling unit using Fisher discriminant analysis," *Energy Convers. Manage.*, vol. 49, no. 12, pp. 3654–3665, Dec. 2008.

[15] Y. Zhang, H. Zhou, S. J. Qin, and T. Chai, "Decentralized fault diagnosis of large-scale processes using multiblock kernel partial least squares," *IEEE Trans. Ind. Informat.*, vol. 6, no. 1, pp. 3–10, Feb. 2010.

[16] A. Moradzadeh, K. Pourhossein, B. Mohammadi-Ivatloo, and F. Mohammadi, "Locating inter-turn faults in transformer windings using isometric feature mapping of frequency response traces," *IEEE Trans. Ind. Informat.*, vol. 17, no. 10, pp. 6962–6970, Oct. 2021.

[17] R. Li and L. Liu, "Research on rolling bearing fault diagnosis based on improved local linear embedding algorithm," in *Proc. Global Rel. Prognostics Health Manage. (PHM-Yantai)*, Oct. 2022, pp. 1–3.

[18] B. Zhang and P. Shang, "Distance correlation entropy and ordinal distance complexity measure: Efficient tools for complex systems," *Nonlinear Dyn.*, vol. 112, no. 2, pp. 1153–1172, Jan. 2024.

[19] S. Wang, Y. Wang, J. Tong, and Y. Chang, "Fault monitoring based on the VLSW-MADF test and DLPPCA for multimodal processes," *Sensors*, vol. 23, no. 2, p. 987, Jan. 2023.

[20] Y.-L. He, Y. Zhao, X. Hu, X.-N. Yan, Q.-X. Zhu, and Y. Xu, "Fault diagnosis using novel AdaBoost based discriminant locality preserving projection with resamples," *Eng. Appl. Artif. Intell.*, vol. 91, May 2020, Art. no. 103631.

[21] G.-F. Lu, Z. Lin, and Z. Jin, "Face recognition using discriminant locality preserving projections based on maximum margin criterion," *Pattern Recognit.*, vol. 43, no. 10, pp. 3572–3579, Oct. 2010.

[22] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[23] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," 2012, *arXiv:1206.5538*.

[24] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2011, pp. 151–161.

[25] Y. Nesterov, "A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$," *Dokl. Akad. Nauk. SSSR*, vol. 269, no. 3, p. 543, 1983.

[26] M. Ranzato, Y.-L. Boureau, and Y. L. Cun, "Sparse feature learning for deep belief networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 20, Dec. 2007, pp. 1185–1192.

[27] H. Akaike, "Akaike's information criterion," in *International Encyclopedia of Statistical Science*. Berlin, Germany: Springer-Verlag, 2011, p. 25.

[28] X. Sha, C. Luo, and N. Diao, "A robust fault detection method based on neighborhood preserving embedding," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–12, 2022.

[29] X. Jin, M. Zhao, T. W. S. Chow, and M. Pecht, "Motor bearing fault diagnosis using trace ratio linear discriminant analysis," *IEEE Trans. Ind. Electron.*, vol. 61, no. 5, pp. 2441–2451, May 2014.

[30] Y.-L. He, K. Li, L.-L. Liang, Y. Xu, and Q.-X. Zhu, "Novel discriminant locality preserving projection integrated with Monte Carlo sampling for fault diagnosis," *IEEE Trans. Rel.*, vol. 72, no. 1, pp. 166–176, Mar. 2023.

[31] N. Zhang, Y. Xu, Q.-X. Zhu, and Y.-L. He, "Farthest-nearest distance neighborhood and locality projections integrated with bootstrap for industrial process fault diagnosis," *IEEE Trans. Ind. Informat.*, vol. 19, no. 5, pp. 6284–6294, May 2023.

[32] J. Gu, Y. Peng, H. Lu, X. Chang, and G. Chen, "A novel fault diagnosis method of rotating machinery via VMD, CWT and improved CNN," *Measurement*, vol. 200, Aug. 2022, Art. no. 111635.

[33] Y. Jin, C. Qin, Z. Zhang, J. Tao, and C. Liu, "A multi-scale convolutional neural network for bearing compound fault diagnosis under various noise conditions," *Sci. China Technol. Sci.*, vol. 65, no. 11, pp. 2551–2563, Nov. 2022.

[34] H. Zhou, W. Chen, L. Cheng, D. Williams, C. W. De Silva, and M. Xia, "Reliable and intelligent fault diagnosis with evidential VGG neural networks," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–12, 2023.

[35] C. Zhang, Q. Guo, and Y. Li, "Fault detection in the Tennessee eastman benchmark process using principal component difference based on K-nearest neighbors," *IEEE Access*, vol. 8, pp. 49999–50009, 2020.

[36] Y. Li, X. Wang, S. Si, and S. Huang, "Entropy based fault classification using the Case Western Reserve University data: A benchmark study," *IEEE Trans. Rel.*, vol. 69, no. 2, pp. 754–767, Jun. 2020.

[37] A. P. Daga, A. Fasana, S. Marchesiello, and L. Garibaldi, "The Politecnico di Turin rolling bearing test rig: Description and analysis of open access data," *Mech. Syst. Signal Process.*, vol. 120, pp. 252–273, Apr. 2019.

[38] A. Lundström and M. O'Nils, "Factory-based vibration data for bearing-fault detection," *Data*, vol. 8, no. 7, p. 115, Jun. 2023.

[39] S. Lundberg and S. Lee, "A unified approach to interpreting model predictions," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, Jan. 2017.

Ruisheng Ran received the B.S. degree in mathematics from Chongqing Normal University, Chongqing, China, in 1998, the M.S. degree in computational mathematics from the University of Electronic Science and Technology of China, Chengdu, China, in 2004, and the Ph.D. degree in computer application technology from the University of Electronic Science and Technology of China, in 2006.

He is currently a Professor with the College of Computer and Information Science, Chongqing Normal University. His research interests include machine learning and computer vision research.

Ting Wang is currently pursuing the master's degree with the College of Computer and Information Science, Chongqing Normal University, Chongqing, China.

Her research interests include machine learning and computer vision research.

Wenfeng Zhang received the Ph.D. degree from the Ocean University of China, Qingdao, China, in 2021.

He is currently a Lecturer with Chongqing Normal University, Chongqing, China. His current research interests include multimedia content analysis and retrieval, computer vision, and deep learning.

Bin Fang received the B.S. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 1989, the M.S. degree in electrical engineering from Sichuan University, Chengdu, China, in 1994, and the Ph.D. degree in electrical engineering from the University of Hong Kong, Hong Kong, in 2001.

He is currently a Professor with the Department of Computer Science, Chongqing University, Chongqing, China. He has published more than 100 technical articles. His research interests include computer vision, pattern recognition, medical image processing, biometrics applications, and document analysis.

Dr. Fang is an Associate Editor of the *International Journal of Pattern Recognition and Artificial Intelligence*.