Data-Driven Fault Diagnosis Using Deep Canonical Variate Analysis and Fisher Discriminant Analysis

Wu, Ping; Lou, Siwei; Zhang, Xujie; He, Jiajun; Liu, Yichao; Gao, Jinfeng

DOI
10.1109/TII.2020.3030179

Publication date
2021

Document Version
Final published version

Published in
IEEE Transactions on Industrial Informatics

Data-Driven Fault Diagnosis Using Deep Canonical Variate Analysis and Fisher Discriminant Analysis

Ping Wu, Siwei Lou, Xujie Zhang, Jiajun He, Yichao Liu, and Jinfeng Gao

***Abstract*—In this article, a novel data-driven fault diagnosis method by combining deep canonical variate analysis and Fisher discriminant analysis (DCVA-FDA) is proposed for complex industrial processes. Inspired by the recently developed deep canonical correlation analysis, a new nonlinear canonical variate analysis (CVA) called DCVA is first developed by incorporating deep neural networks into CVA. Based on DCVA, a residual generator is designed for the fault diagnosis process. FDA is applied in the feature space spanned by residual vectors. Then, a Bayesian inference classifier is performed in the reduced dimensional space of FDA to label the class of process data. A continuous stirred-tank reactor and an industrial benchmark of the Tennessee Eastman process are carried out to test the performance of DCVA-FDA fault diagnosis. The experimental results demonstrate that the proposed DCVA-FDA fault diagnosis is able to significantly improve the fault diagnosis performance when compared to other methods also examined in this article.**

***Index Terms*—Bayesian classifier, canonical variate analysis (CVA), deep neural networks (DNN), fault diagnosis, Fisher discriminant analysis (FDA), residual generator.**

List of Abbreviations

CCA	Canonical correlation analysis.
CVA	Canonical variate analysis.
CVA-FDA	Canonical variate analysis-Fisher discriminant analysis.
CSTR	Continuous stirred-tank reactor.
DCCA	Deep canonical correlation analysis.
DCVA	Deep canonical variate analysis.
DCVA-FDA	Deep canonical variate analysis-Fisher discriminant analysis.
DFDA	Dynamic Fisher discriminant analysis.
DNN	Deep neural networks.
FDA	Fisher discriminant analysis.
KFDA	Kernel Fisher discriminant analysis.
KDFDA	Kernel dynamic Fisher discriminant analysis.
kNN	k-nearest neighbors.
L-BFGS	Limited-memory Broyden–Fletcher–Goldfarb–Shanno.
MVA	Multivariate analysis.
PCA	Principal component analysis.
PLS	Partial least squares.
ReLU	Rectified linear unit.
SGD	Stochastic gradient descent.
SVD	Singular value decomposition.
TEP	Tennessee eastman process.

I. Introduction

DUE TO the ever-increasing demands on high levels of plant safety and product quality, fault diagnosis plays a vital role in reducing system downtime and improving product quality for complex industrial processes [1]. Fault diagnosis is capable of detecting the process abnormality and classifying the abnormality types to locate the root causes of the observed out-of-control status [2].

Broadly speaking, fault diagnosis techniques can be categorized into three main classes of model-based, expert knowledge, and data-driven methods [3]–[6]. Model-based methods rely on an accurate mathematical model. However, the mathematical model, which takes modeling errors and uncertainties into account is very difficult to obtain for complex industrial processes. For expert knowledge-based approaches, fault diagnosis is implemented by capturing knowledge to draw conclusions in a formal methodology. The limitation of expert knowledge-based methods is that the knowledge base is not easily extracted from past experiences with the process due to the complexity of modern industrial processes. Different from model-based and expert knowledge-based approaches, data-driven based fault diagnosis only needs to use a large amount of historical data. In data-driven methods, fault diagnosis is carried out by applying machine learning techniques to process data [7]. Data-driven methods treat fault diagnosis as a multilabel classification task [8].

Manuscript received July 10, 2020; revised August 17, 2020; accepted October 6, 2020. Date of publication October 12, 2020; date of current version February 22, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61703371 and in part by the Social Development Project of Zhejiang Provincial Public Technology Research under Grant LGF19F030004. Paper no. TII-20-3359. *(Corresponding author: Ping Wu.)*

Ping Wu, Siwei Lou, Xujie Zhang, Jiajun He, and Jinfeng Gao are with the Faculty of Mechanical Engineering and Automation, Zhejiang Sci-Tech University, Hangzhou 310018, China (e-mail: pingwu@zstu.edu.cn; 17826856306@163.com; 201920505009@mails.zstu.edu.cn; hejiajun_dg@163.com; gaojf@zstu.edu.cn).

Yichao Liu is with the Delft Center for Systems and Controls, Delft University of Technology, CD 2628 Delft, The Netherlands (e-mail: Y.Liu-17@tudelft.nl).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2020.3030179

With the rapid development of information and communication technology, a large amount of industrial process data can be efficiently collected and processed. Therefore, data-driven fault diagnosis plays a critical role in complex industrial processes due to its practicability and efficiency.

As a representative class of data-driven methods, multivariate analysis (MVA) such as principal component analysis (PCA), partial least squares (PLS), canonical variate analysis (CVA), Fisher discriminant analysis (FDA) have recently been widely applied for process monitoring and fault diagnosis in a variety of industrial processes [9]–[13]. Compared to other MVA methods such as PCA and PLS, FDA is a well-known technique for supervised classification and dimensionality reduction. In FDA, a set of projection vectors is determined to simultaneously maximize the between-class scatter and minimize the within-class scatter. Due to its simplicity and practicability, FDA has become one of the most popular tools to conduct fault diagnosis for industrial processes.

For complex industrial processes, dynamics and nonlinearity are commonly observed in process data [14], [15]. To address the dynamics in process data, FDA can be extended to dynamic Fisher discriminant analysis (DFDA) by augmenting the process variables to time-lagged vectors [16]. The main shortcoming of DFDA is that the dynamic characteristic of process data cannot be fully explored by time-lagged extension. And the interpretability of the data stacking strategy is poor [17]. On the other hand, to handle the nonlinearity of process data, kernel Fisher discriminant analysis (KFDA) was developed by introducing kernel methods [18]. The basic idea of KFDA is to map the process variables into a high-dimensional space through kernel functions and then perform the linear FDA in the kernel feature space. A number of fault diagnosis methods have been developed to tackle specific problems by utilizing KFDA. Ge *et al.* [19] proposed a semisupervised KFDA method for nonlinear fault classification with labeled and unlabeled data. Feng *et al.* [20] integrated the local and global discriminant information into KFDA to address the issues of non-Gaussianity and nonlinearity. Additionally, KFDA can be extended to kernel dynamic FDA (KDFDA) to take both dynamics and nonlinearity into consideration by performing KFDA on the time-lagged extension of the original data [21]. However, kernel methods have some limitations, e.g., difficulty in determining parameters such as kernel width and number of features, and heavy computational burden related to the calculation of a kernel function for online samples [22].

CVA is a further MVA method and recognized as an efficient approach in process monitoring and fault diagnosis for dynamic systems. CVA is originated from canonical correlation analysis (CCA). In CVA, the correlation between the "past" data and "future" data is analyzed through CCA. CVA takes into account serial correlations in the process data, which makes it more efficient in fault detection for dynamic systems [23]–[25]. Odiowei and Cao [12] applied CVA to determine the state variables directly from the process measurements, where the correlations between the past values of measurements and the future values of measurements are considered. Chen *et al.* [26] utilized CCA to maximize the correlation between the combination of the past values of process inputs and outputs and the combination of the future values of outputs. Due to its superiority, the combination of CVA with other MVA methods can largely improve the process monitoring and fault diagnosis performance. Zhang *et al.* [27] proposed a combined strategy of CVA and slow feature analysis to monitor process dynamics resulting from closed-loop control by examining both serial correlations and variation speed of process data. Jiang *et al.* [17] developed a fault diagnosis method by combining CVA with FDA to enhance the performance of fault diagnosis, which applies CVA to obtain the state space vector for pretreating the process data and subsequently utilizes FDA for fault classification. However, the issue of nonlinearity is not addressed in these combination methods.

Deep neural networks (DNN) methods such as deep Boltzmann machines [28], deep auto-encoders [29], and deep nonlinear feed-forward networks [30], have attained an empirical success on a wide variety of tasks. By utilizing DNN, deep learning methods overcome the shortcoming of kernel-based methods that the representation is limited by the fixed kernel. Besides, DNN is a parametric method to extract the nonlinear features, rather than a nonparametric method such as kernel methods. Thus, the time required to compute the representations of new data points does not scale with the size of the training dataset in DNN. Jiang and Yan [22] proposed a regularized deep correlated representation method that incorporates deep belief networks and CCA for nonlinear process monitoring. Recently, deep canonical correlation analysis (DCCA) was proposed to learn complex nonlinear transformations of two views of data based on DNN, such that the resulting representations are highly linearly correlated [31]. The basic idea behind DCCA is to compute representations of the two views by passing them through multiple stacked layers of nonlinear transformation. DCCA has been successfully applied in a number of case studies such as image and text processing, acoustic features learning, and electroencephalography signal processing [32]–[34]. However, to the best of author's knowledge, the use of MVA based on DNN has been rarely investigated in fault diagnosis so far.

To address the issues of dynamics and nonlinearity of data and pursue improved fault diagnosis performance, a novel data-driven fault diagnosis method is proposed by combining deep canonical variate analysis (DCVA) and FDA. We develop a new nonlinear CVA called DCVA to handle the nonlinear and dynamic characteristics of industrial process data. In DCVA, the nonlinear transformations of past and future data vectors are learned through DNN so that the correlation between nonlinear transformations is maximized. Then, a residual generator is established from DCVA features for the fault diagnosis process. Since the relationship between DCVA features can be treated as linear, FDA is performed on the residual vectors to classify faults. The proposed method is referred to as DCVA-FDA in this article.

The main contributions of this article can be summarized as follows.

1) A new nonlinear CVA is developed by borrowing the idea of DCCA to cope with the dynamics and nonlinearity of industrial process data.

2) A more efficient approach for extracting discriminant features is proposed by combining the developed DCVA and FDA for fault diagnosis.
3) An improved performance is provided by the proposed DCVA-FDA fault diagnosis through experiments on a continuous stirred-tank reactor process (CSTR) and an industrial benchmark of Tennessee Eastman process.

The rest of this article is organized as follows. In Section II, the preliminaries of CVA-based residual generator are briefly reviewed. Then, in Section III, DCVA and its application in residual generator are elaborated. In Section IV, the DCVA-FDA fault diagnosis scheme is developed. A CSTR process and an industrial benchmark of Tennessee Eastman process (TEP) are used to show its feasibility and effectiveness in Section V. Finally, Section VI concludes this article.

II. CVA-Based Residual Generator Revisited

Let $\boldsymbol{u}(k) \in \mathbb{R}^m$ and $\boldsymbol{y}(k) \in \mathbb{R}^l$ be the input and output at time instant k, respectively. The past data vector $\boldsymbol{z}_p(k) \in \mathbb{R}^{(m+l)q}$ is constructed by augmenting the "past" inputs $\boldsymbol{u}(k)$ and outputs $\boldsymbol{y}(k)$, and the future data vector $\boldsymbol{y}_f(k) \in \mathbb{R}^{lq}$ is formed by augmenting the "future" outputs $\boldsymbol{y}(k)$,

$$\boldsymbol{z}_p(k) = \begin{bmatrix}\boldsymbol{u}(k-1)^T & \cdots & \boldsymbol{u}(k-q)^T & \boldsymbol{y}(k-1)^T \\ & \cdots & \boldsymbol{y}(k-q)^T\end{bmatrix}^T$$

$$\boldsymbol{y}_f(k) = \begin{bmatrix}\boldsymbol{y}(k)^T & \boldsymbol{y}(k+1)^T & \cdots & \boldsymbol{y}(k+q-1)^T\end{bmatrix}^T$$

where q is the number of time lags in the past and future data vectors. q is often determined through checking the auto-correlation of the process variables. Assumed that a training dataset with N samples of $(\boldsymbol{u}(k), \boldsymbol{y}(k), k = 1, \ldots, N)$ is collected, then the Hankel matrices are formed by the normalized past and future data vectors,

$$\boldsymbol{Z}_p = \begin{bmatrix}\boldsymbol{z}_p(q+1) & \boldsymbol{z}_p(q+2) & \cdots & \boldsymbol{z}_p(M)\end{bmatrix} \in \mathbb{R}^{(m+l)q \times M}$$

$$\boldsymbol{Y}_f = \begin{bmatrix}\boldsymbol{y}_f(q+1) & \boldsymbol{y}_f(q+2) & \cdots & \boldsymbol{y}_f(M)\end{bmatrix} \in \mathbb{R}^{lq \times M}$$

where $M = N - 2q + 1$.

The objective of CVA is to seek for the pairs of linear projections $(\boldsymbol{J}^T\boldsymbol{z}_p(k), \boldsymbol{L}^T\boldsymbol{y}_f(k))$ of the past and future data vectors so that these projections are maximally correlated

$$\begin{aligned} \max_{\boldsymbol{J},\boldsymbol{L}} \quad & \frac{1}{M-1} tr(\boldsymbol{J}^T\boldsymbol{Z}_p\boldsymbol{Y}_f^T\boldsymbol{L}) \\ \text{s.t.} \, & \boldsymbol{J}^T\boldsymbol{J} = \boldsymbol{I} \\ & \boldsymbol{L}^T\boldsymbol{L} = \boldsymbol{I} \end{aligned} \tag{1}$$

where $\boldsymbol{J}$ and $\boldsymbol{L}$ are the weighting matrices to be determined. Here, $\boldsymbol{I}$ is a unity matrix with appropriate dimensions.

The optimization problem (1) can be readily solved by performing singular value decomposition (SVD). First, the sample covariance matrices $(\boldsymbol{\Sigma}_{pp}, \boldsymbol{\Sigma}_{ff})$ and crosscovariance matrix $\boldsymbol{\Sigma}_{pf}$ are estimated as follows:

$$\boldsymbol{\Sigma}_{pp} = \frac{1}{M-1}\boldsymbol{Z}_p\boldsymbol{Z}_p^T \tag{2}$$

$$\boldsymbol{\Sigma}_{ff} = \frac{1}{M-1}\boldsymbol{Y}_f\boldsymbol{Y}_f^T \tag{3}$$

$$\boldsymbol{\Sigma}_{pf} = \frac{1}{M-1}\boldsymbol{Z}_p\boldsymbol{Y}_f^T. \tag{4}$$

Then, the Hankel matrix $\mathcal{H}$ is decomposed through SVD operation

$$\mathcal{H} = \boldsymbol{\Sigma}_{pp}^{-1/2}\boldsymbol{\Sigma}_{pf}\boldsymbol{\Sigma}_{ff}^{-1/2} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T \tag{5}$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are the left and right singular vectors, respectively. $\boldsymbol{S}$ is a diagonal matrix, which consists of singular values. If the rank of $\mathcal{H}$ is n, then $\boldsymbol{S} = \text{diag}(\delta_1, \delta_2, \ldots, \delta_n, 0, 0, \ldots, 0), \delta_1 > \delta_2 > \cdots > \delta_n$.

From the result of SVD, the weight matrices $\boldsymbol{J}$ and $\boldsymbol{L}$, which comprise the n weight vectors can be constructed from the left and right singular vectors of $\mathcal{H}$ as follows:

$$\boldsymbol{J} = \boldsymbol{\Sigma}_{pp}^{-1/2}\boldsymbol{U}(:, 1:n) \tag{6}$$

$$\boldsymbol{L} = \boldsymbol{\Sigma}_{ff}^{-1/2}\boldsymbol{V}(:, 1:n). \tag{7}$$

In the CVA-based fault detection approach, the residual generator can be established from CVA features $\boldsymbol{L}^T\boldsymbol{y}_f(k)$ and $\boldsymbol{J}^T\boldsymbol{z}_p(k)$ [24], [26]

$$\boldsymbol{r}(k) = \boldsymbol{L}^T\boldsymbol{y}_f(k) - \boldsymbol{S}_n\boldsymbol{J}^T\boldsymbol{z}_p(k) \tag{8}$$

where the diagonal matrix $\boldsymbol{S}_n$ is constructed by n singular values $\boldsymbol{S}_n = \text{diag}(\delta_1, \delta_2, \ldots, \delta_n)$. Usually, a Hotelling's statistic is established based on the residual vector $\boldsymbol{r}(k)$ to monitor the process status [25].

Remark 1: The state vector can be derived as $\boldsymbol{x}(k) = \boldsymbol{J}^T\boldsymbol{z}_p(k)$ [35]. A Hotelling's statistic which is built from the estimated $\boldsymbol{x}(k)$ can also be utilized for monitoring the process status [12], [35]. In addition, a CVA-FDA method was developed by applying FDA on the estimated $\boldsymbol{x}(k)$ for fault diagnosis in [17]. As described in [24], the statistic formed by $\boldsymbol{r}(k)$ is more sensitive than the one from state vector. Therefore, we adopt this type of residual generator for the purpose of fault diagnosis in this article.

III. Proposed DCVA

In CVA-based fault detection methods, the core is to investigate the relationship between past and future data vectors. Representations of the past and future data vectors are learned by utilizing CCA to maximize the correlation between past and future data vectors. Then, a residual generator is built by using these representations for fault detection. However, nonlinear characteristics are commonly exhibited in industrial processes. CVA cannot deal with the nonlinearity of process data. Recently, DCCA was developed for learning the nonlinear representations of two views by integrating DNN into CCA [33]. Different from kernel-based methods, DCCA is a parametric method. It does not require an inner product operation. The whole training data are not involved in computing the representations of the new data samples. It has been proved that the DCCA can achieve superior performance over kernel-based methods [31]. Inspired by the idea of DCCA, we develop DCVA to address the issues of dynamics and nonlinearity for fault diagnosis of complex industrial processes in this article.

First, representations of the past data vector $\boldsymbol{z}_p$ and future data vector $\boldsymbol{y}_f$ are computed through multiple stacked layers of nonlinear transformation. For simplicity, we assume that each intermediate layer in the neural networks for the past data vector $\boldsymbol{z}_p$ has c_1 units, and the final layer which is also the output layer has o units. For instance, the outputs of the first layer of $\boldsymbol{z}_p$ are $\boldsymbol{h}_1 = s(\boldsymbol{W}_p^1 \boldsymbol{z}_p + \boldsymbol{b}_p^1) \in \mathbb{R}^{c_1}$. Here, $\boldsymbol{W}_p^1 \in \mathbb{R}^{c_1 \times (m+l)q}$ is a weight matrix, and $\boldsymbol{b}_1^1 \in \mathbb{R}^{c_1}$ is a bias vector, and $s(\cdot)$ is a component-wise nonlinear function. The outputs of the next layer are computed from the preceding outputs $\boldsymbol{h}_1$ as $\boldsymbol{h}_2 = s(\boldsymbol{W}_p^2 \boldsymbol{h}_1 + \boldsymbol{b}_p^2) \in \mathbb{R}^{c_1}$. Followed by the same procedure, the final representation $\boldsymbol{f}_p(\boldsymbol{z}_p) = s(\boldsymbol{W}_p^d \boldsymbol{h}_{d-1} + \boldsymbol{b}_p^d) \in \mathbb{R}_p^o$ is computed. For the convenience of optimization procedure, biases $\boldsymbol{b}$ at each layer are usually included in the weight matrix $\boldsymbol{W}_p$ by appending an extra 1 to its input. Likewise, the representation $\boldsymbol{f}_f(\boldsymbol{y}_f)$ of the future data vector $\boldsymbol{y}_f$ is obtained with different parameters $\boldsymbol{W}_f$.

The main goal of DCVA is to jointly learn parameters $\boldsymbol{W}_p, \boldsymbol{W}_f$ such that the correlation between $\boldsymbol{f}_p(\boldsymbol{z}_p)$ and $\boldsymbol{f}_f(\boldsymbol{y}_f)$ is maximized

$$\begin{aligned} \max_{\boldsymbol{W}_p, \boldsymbol{W}_f, \mathcal{J}, \mathcal{L}} \quad & \frac{1}{M-1} tr(\mathcal{J}^T \boldsymbol{F}_p(\boldsymbol{Z}_p; \boldsymbol{W}_p) \boldsymbol{F}_f(\boldsymbol{Y}_f; \boldsymbol{W}_f) \mathcal{L}) \\ \text{s.t.} \quad & \mathcal{J}^T \mathcal{J} = \boldsymbol{I} \\ & \mathcal{L}^T \mathcal{L} = \boldsymbol{I} \end{aligned} \tag{9}$$

where $\boldsymbol{F}_p(\boldsymbol{Y}_p; \boldsymbol{W}_p)$ and $\boldsymbol{F}_f(\boldsymbol{Y}_f; \boldsymbol{W}_f)$ are the representations of the past data and future data vectors for all the training dataset.

Generally, appropriate regularization terms are included to deal with numerical problems and reduce the detection of spurious correlations [31]. Therefore, the constraints in (9) are rewritten by imposing regularization terms as follows:

$$\begin{cases} \mathcal{J}^T(\frac{1}{M-1} \boldsymbol{F}_p(\boldsymbol{Z}_p; \boldsymbol{W}_p) \boldsymbol{F}_p(\boldsymbol{Z}_p; \boldsymbol{W}_p)^T + \lambda_p \boldsymbol{I}) \mathcal{J} = \boldsymbol{I} \\ \mathcal{L}^T(\frac{1}{M-1} \boldsymbol{F}_f(\boldsymbol{Y}_f; \boldsymbol{W}_f) \boldsymbol{F}_f(\boldsymbol{Y}_f; \boldsymbol{W}_f)^T + \lambda_f \boldsymbol{I}) \mathcal{L} = \boldsymbol{I} \end{cases} \tag{10}$$

where λ_p and λ_f are the regularization coefficients.

For the typical DNNs used in regression or classification, the optimization problems are usually without constraints. Besides, the objective functions can be expressed as the expectation (or sum) of error functions (e.g., squared loss or crossentropy). However, there are two networks in maximizing the correlation as shown in(9). Andrew *et al.* [31] applied a full batch algorithm, (i.e., Limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm, L-BFGS), to solve the optimization problem. To improve the computational efficiency, an efficient stochastic gradient descent (SGD) algorithm was developed by Wang *et al.* [33]. The results showed that the stochastic training approach can produce both faster training and better performing features than L-BFGS for solving the optimization problem (9) [33]. In this article, we also adopt the SGD algorithm presented in [33] to train the networks of DCVA.

It is noting that $\mathcal{J}$ and $\mathcal{L}$ have a closed-form solution for fixed functions $\boldsymbol{f}_f$ and $\boldsymbol{f}_p$. Therefore, we can obtain

$$\mathcal{T} = \tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2} \tilde{\boldsymbol{\Sigma}}_{pf} \tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2} = \mathcal{U} \mathcal{S} \mathcal{V}^T \tag{11}$$

where $\tilde{\boldsymbol{\Sigma}}_{pp} = \frac{1}{M-1} \boldsymbol{F}_p(\boldsymbol{Z}_p; \boldsymbol{W}_p) \boldsymbol{F}_p(\boldsymbol{Z}_p; \boldsymbol{W}_p)^T + \lambda_p \boldsymbol{I}$, $\tilde{\boldsymbol{\Sigma}}_{ff} = \frac{1}{M-1} \boldsymbol{F}_f(\boldsymbol{Y}_f; \boldsymbol{W}_f) \boldsymbol{F}_f(\boldsymbol{Y}_f; \boldsymbol{W}_f)^T + \lambda_f \boldsymbol{I}$, and $\tilde{\boldsymbol{\Sigma}}_{pf} = \frac{1}{M-1} \boldsymbol{F}_p(\boldsymbol{Z}_p; \boldsymbol{W}_p) \boldsymbol{F}_f(\boldsymbol{Y}_f; \boldsymbol{W}_f)^T$. $\mathcal{U}$ is constructed by the left singular vectors corresponding to the singular values, which are arranged in descending order of their magnitudes. $\mathcal{V}$ is formed in a similar way. Then, $\mathcal{J}$ and $\mathcal{L}$ can be obtained as $\mathcal{J} = \tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2} \mathcal{U}$ and $\mathcal{L} = \tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2} \mathcal{V}$, respectively.

Substituting $\mathcal{J}$ and $\mathcal{L}$ into (9),

$$tr(\mathcal{J}^T \boldsymbol{F}_p(\boldsymbol{Z}_p; \boldsymbol{W}_p) \boldsymbol{F}_f(\boldsymbol{Y}_f; \boldsymbol{W}_f) \mathcal{L}) = \Sigma_{j=1}^L \sigma_j(\mathcal{T}). \tag{12}$$

Denote $\tilde{\mathcal{T}} = \tilde{\mathcal{U}} \mathcal{S}_L \tilde{\mathcal{V}}^T$ as the rank-L SVD of $\mathcal{T}$. $\mathcal{S}_L = \text{diag}(\sigma_1(\tilde{\mathcal{T}}), \sigma_2(\tilde{\mathcal{T}}), \ldots, \sigma_L(\tilde{\mathcal{T}}))$ where $\sigma_1(\tilde{\mathcal{T}}) > \sigma_2(\tilde{\mathcal{T}}) > \cdots > \sigma_L(\tilde{\mathcal{T}})$ are the L largest singular values of $\tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2} \tilde{\boldsymbol{\Sigma}}_{pf} \tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2}$. $\tilde{\mathcal{U}} = \mathcal{U}(:, 1:L)$ and $\tilde{\mathcal{V}} = \mathcal{V}(:, 1:L)$ are formed by the first L singular vectors of $\mathcal{U}$ and $\mathcal{V}$, respectively. The gradient of the total correlations with respect to the feature matrix $\boldsymbol{F}_p$ is

$$\frac{\partial \Sigma_{j=1}^L \sigma_j(\tilde{\mathcal{T}})}{\partial \boldsymbol{F}_p} = \frac{1}{M-1}(2\Delta_{11} \boldsymbol{F}_p + \Delta_{12} \boldsymbol{F}_f). \tag{13}$$

Here, $\Delta_{11} = -\frac{1}{2} \tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2} \tilde{\mathcal{U}} \mathcal{S}_L \tilde{\mathcal{U}}^T \tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2}$, $\Delta_{12} = -\tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2} \tilde{\mathcal{U}} \tilde{\mathcal{V}}^T \tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2}$. In the same manner, $\partial \Sigma_{j=1}^L \sigma_j(\tilde{\mathcal{T}}) / \partial \boldsymbol{F}_f$ has a symmetric expression

$$\frac{\partial \Sigma_{j=1}^L \sigma_j(\tilde{\mathcal{T}})}{\partial \boldsymbol{F}_f} = \frac{1}{M-1}(2\Delta_{21} \boldsymbol{F}_f + \Delta_{22} \boldsymbol{F}_p) \tag{14}$$

where $\Delta_{21} = -\frac{1}{2} \tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2} \tilde{\mathcal{V}} \mathcal{S}_L \tilde{\mathcal{V}}^T \tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2}$, $\Delta_{22} = -\tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2} \tilde{\mathcal{V}} \tilde{\mathcal{U}}^T \tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2}$. The details of the derivation of gradients can be found in [31].

The update of $\boldsymbol{W}_p$ and $\boldsymbol{W}_f$ can be computed through standard backpropagation. Denote $\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_p & \boldsymbol{W}_f \end{bmatrix}$. Based on the derived gradients (13) and (14), the gradient $\nabla_{\boldsymbol{W}}$ can be readily obtained for a determined neural networks architecture. Given the gradient $\nabla_{\boldsymbol{W}}$ with respect to the all weight parameters evaluated on minibatches where the size of minibatch is s_{mb}, the weight parameters $\boldsymbol{W}$ at step t are updated as follows:

$$\begin{aligned} \Delta \boldsymbol{W}^t &= \mu^t \Delta \boldsymbol{W}^{t-1} - \epsilon^t \nabla_{\boldsymbol{W}} \\ \boldsymbol{W}^t &= \boldsymbol{W}^{t-1} + \Delta \boldsymbol{W}^t \end{aligned} \tag{15}$$

where $\mu^t \in [0, 1)$ and ϵ^t are the momentum parameter and learning rate at step t, respectively. In this article, we used fixed μ and ϵ in the SGD algorithm. Based on the derived gradients and update rules, $\boldsymbol{W}$ is learned through the SGD algorithm until the total correlation stops improving on a held-out validation set. The outline of DCVA is shown in Algorithm 1.

Algorithm 1: DCVA.

```
Input: Data matrix $\boldsymbol{Z}_p \in \mathbb{R}^{(m+l)q \times M}, \boldsymbol{Y}_f \in \mathbb{R}^{lq \times M}$.
 Learning rate $\epsilon$, momentum parameter $\mu$, size of
 minibatch $s_{mb}$, regularization parameters $\lambda_p$ and $\lambda_f$.
 Initialization $\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_p & \boldsymbol{W}_f \end{bmatrix}$. Maximum number of
 epoch $N_{me}$.
Output: The updated $\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_p & \boldsymbol{W}_f \end{bmatrix}$, $\tilde{\mathcal{J}}$, $\tilde{\mathcal{L}}$ and $\mathcal{S}_L$.
for $t \leftarrow 1$ to $N_{me}$ do
  Randomly choose a minibatch $\boldsymbol{Z}_{pt}, \boldsymbol{Y}_{ft}$.
  Compute the matrices
    $\tilde{\boldsymbol{\Sigma}}_{pp} = \frac{1}{M-1}\boldsymbol{F}_p(\boldsymbol{Z}_{pt};\boldsymbol{W}_p)\boldsymbol{F}_p(\boldsymbol{Z}_{pt};\boldsymbol{W}_p)^T + \lambda_p \boldsymbol{I}$
    $\tilde{\boldsymbol{\Sigma}}_{ff} = \frac{1}{M-1}\boldsymbol{F}_f(\boldsymbol{Y}_{ft};\boldsymbol{W}_f)\boldsymbol{F}_f(\boldsymbol{Y}_{ft};\boldsymbol{W}_f)^T + \lambda_f \boldsymbol{I}$
    $\tilde{\boldsymbol{\Sigma}}_{pf} = \frac{1}{M-1}\boldsymbol{F}_p(\boldsymbol{Z}_{pt};\boldsymbol{W}_p)\boldsymbol{F}_f(\boldsymbol{Y}_{ft};\boldsymbol{W}_f)^T$
  Perform SVD
    $\mathcal{T} = \tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2}\tilde{\boldsymbol{\Sigma}}_{pf}\tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2} = \mathcal{U}\mathcal{S}\mathcal{V}^T$
  Calculate the rank-$L$ SVD of $\mathcal{T}$
    $\tilde{\mathcal{T}} = \tilde{\mathcal{U}}\mathcal{S}_L\tilde{\mathcal{V}}^T$
  Compute the gradient
    $\frac{\partial \Sigma_{j=1}^{L}\sigma_j(\tilde{\mathcal{T}})}{\partial \boldsymbol{F}_p} = \frac{1}{M-1}(2\Delta_{11}\boldsymbol{F}_p + \Delta_{12}\boldsymbol{F}_f)$
    $\frac{\partial \Sigma_{j=1}^{L}\sigma_j(\tilde{\mathcal{T}})}{\partial \boldsymbol{F}_f} = \frac{1}{M-1}(2\Delta_{21}\boldsymbol{F}_f + \Delta_{22}\boldsymbol{F}_p)$
  Calculate $\nabla_{\boldsymbol{W}}$ according to $\frac{\partial \Sigma_{j=1}^{L}\sigma_j(\tilde{\mathcal{T}})}{\partial \boldsymbol{F}_p}$ and $\frac{\partial \Sigma_{j=1}^{L}\sigma_j(\tilde{\mathcal{T}})}{\partial \boldsymbol{F}_f}$
  Update $\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_p & \boldsymbol{W}_f \end{bmatrix}$
    $\Delta \boldsymbol{W}^t = \mu^t \Delta \boldsymbol{W}^{t-1} - \epsilon^t \nabla_{\boldsymbol{W}}$
    $\boldsymbol{W}^t = \boldsymbol{W}^{t-1} + \Delta \boldsymbol{W}^t$
end for
Calculate the weigh matrices
    $\tilde{\mathcal{J}} = \tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2}\tilde{\mathcal{U}}$
    $\tilde{\mathcal{L}} = \tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2}\tilde{\mathcal{V}}$
```

Based on the updated $\boldsymbol{W}$, the weight matrices $\tilde{\mathcal{L}}$ and $\tilde{\mathcal{J}}$, which are corresponding to the largest L singular values can be obtained

$$\tilde{\mathcal{J}} = \tilde{\boldsymbol{\Sigma}}_{pp}^{-1/2}\tilde{\mathcal{U}} \tag{16}$$

$$\tilde{\mathcal{L}} = \tilde{\boldsymbol{\Sigma}}_{ff}^{-1/2}\tilde{\mathcal{V}}. \tag{17}$$

Similarly, the residual generator $\tilde{\boldsymbol{r}} \in \mathbb{R}^L$ is established from the DCVA features $\tilde{\mathcal{L}}^T\boldsymbol{f}_f(\boldsymbol{y}_f(k))$ and $\tilde{\mathcal{J}}^T\boldsymbol{f}_p(\boldsymbol{z}_p(k))$ for fault diagnosis

$$\tilde{\boldsymbol{r}}(k) = \tilde{\mathcal{L}}^T\boldsymbol{f}_f(\boldsymbol{y}_f(k)) - \mathcal{S}_L\tilde{\mathcal{J}}^T\boldsymbol{f}_p(\boldsymbol{z}_p(k)). \tag{18}$$

TABLE I
HYPERPARAMETERS IN DCVA MODEL

Parameter	Description	Suggested range
μ	momentum parameter	$[0,1)$
ϵ	learning rate	$(0,1)$
λ_p	regularization parameter	$(0,1)$
λ_f	regularization parameter	$(0,1)$
s_{mb}	size of minibatch	$[(m+l)q, M]$

Remark 2: The determination of the width and depth of DNN is a challenging work in deep learning. In this article, the numbers of neurons and layers are determined by cross-validation. In general, more neurons and layers would provide improved performance [36]. However, there is a tradeoff between computational complexity and performance. Basically, in this article, the number of neurons per layer is selected to be slightly larger than the dimensions of $\boldsymbol{z}_p$ and $\boldsymbol{y}_f$. Similar to CVA [12], the DCVA dimension L is selected according to the singular values of (11) in the proposed DCVA. In the case studies, the DCVA dimension L is selected as a smaller value than the number of neurons of the hidden layer, as highly correlated representations of $\boldsymbol{z}_p$ and $\boldsymbol{y}_f$ are obtained through DCVA. For DCVA, Table I lists these hyperparameters to be determined.

Since it is difficult to establish the quantitative relationship between the performance of fault diagnosis and these hyperparameters, a common way to select the appropriate structure is through cross-validation. To decide hyperparameters for DCVA, cross-validation by grid search is adopted in this article.

IV. PROPOSED FAULT DIAGNOSIS METHOD

For fault diagnosis, discriminant analysis is consequently performed in the feature space formed by the residual vectors $\tilde{\boldsymbol{r}}(k)$. Due to the relationship between the DCVA features is linear, the residual vectors $\tilde{\boldsymbol{r}}(k)$ can be analyzed by traditional FDA. FDA is a typical pattern classification method [16]. The basic idea behind FDA is to optimize the Fisher criterion by determining a set of projection vectors.

Assumed that N_{all} observations of residual vectors $\tilde{\boldsymbol{r}}(k)$ are computed from fault-free and $c-1$ faulty classes. To elaborate the implementation of FDA, we denote the training data matrix $\tilde{\boldsymbol{R}} \in \mathbb{R}^{N_{\text{all}} \times L}$ which consists of N_{all} residual vectors $\tilde{\boldsymbol{r}}(k)$. The within-scatter matrix $\boldsymbol{S}_{w,j}$ is defined for the jth class $\mathbb{R}_j$,

$$\boldsymbol{S}_{w,j} = \sum_{\tilde{\boldsymbol{r}}(k) \in \mathbb{R}_j} (\tilde{\boldsymbol{r}}(k) - \tilde{\boldsymbol{r}}_{m,j})(\tilde{\boldsymbol{r}}(k) - \tilde{\boldsymbol{r}}_{m,j})^T \tag{19}$$

where $\tilde{\boldsymbol{r}}_{m,j}$ is the mean vector of the jth class $\mathbb{R}_j$. By summing all within-scatter matrix $\boldsymbol{S}_{w,j}$, the within-class-scatter matrix $\boldsymbol{S}_w$ is then derived as follows:

$$\boldsymbol{S}_w = \sum_{j=1}^{c} \boldsymbol{S}_{w,j}. \tag{20}$$

The total-scatter matrix $\boldsymbol{S}_t$ is given by

$$\boldsymbol{S}_t = \sum_{\tilde{\boldsymbol{r}}(k)\in\tilde{\boldsymbol{R}}} (\tilde{\boldsymbol{r}}(k) - \tilde{\boldsymbol{r}}_m)(\tilde{\boldsymbol{r}}(k) - \tilde{\boldsymbol{r}}_m)^T \tag{21}$$

where $\tilde{\boldsymbol{r}}_m$ is the mean vector of $\tilde{\boldsymbol{R}}$.

The between-class-scatter matrix $\boldsymbol{S}_b$ is then calculated as follows:

$$\boldsymbol{S}_b = \sum_{j=1}^{c} n_j(\tilde{\boldsymbol{r}}_m - \tilde{\boldsymbol{r}}_{m,j})(\tilde{\boldsymbol{r}}_m - \tilde{\boldsymbol{r}}_{m,j})^T = \boldsymbol{S}_t - \boldsymbol{S}_w. \tag{22}$$

Here, n_j is the observation number of jth class.

In FDA, the Fisher criterion is used to indicate the ratio of the projection variance (scatter) of means of classes to the projection variance (scatter) of class instances. Therefore, the goal of FDA can be set to derive the optimal discriminant directions $\tilde{\boldsymbol{w}}_i, i = 1, \ldots, d, d \leq c - 1$, which maximize the Fisher criterion $\theta(\tilde{\boldsymbol{w}})$

$$\max_{\tilde{\boldsymbol{w}}} \theta(\tilde{\boldsymbol{w}}) := \frac{\tilde{\boldsymbol{w}}^T \boldsymbol{S}_b \tilde{\boldsymbol{w}}}{\tilde{\boldsymbol{w}}^T \boldsymbol{S}_w \tilde{\boldsymbol{w}}}. \tag{23}$$

The optimization problem (23) is equivalent to a generalized eigenvalue problem

$$\boldsymbol{S}_b \tilde{\boldsymbol{w}}_i = \mu_i \boldsymbol{S}_w \tilde{\boldsymbol{w}}_i \tag{24}$$

where μ_i is the generalized eigenvalue and the Fisher discriminant direction $\tilde{\boldsymbol{w}}_i$ is the FDA eigenvector.

From the result of generalized eigenvalue problem (24), the optimal discriminant directions $\tilde{\boldsymbol{W}}_b$ can be derived by selecting d FDA vectors $\tilde{\boldsymbol{w}}_i, i = 1, \ldots, d$. Then, the discriminant vector $\tilde{\boldsymbol{d}}(k)$, which is also the projection of $\tilde{\boldsymbol{r}}(k)$ onto the discriminant subspace can be represented below

$$\tilde{\boldsymbol{d}}(k) = \tilde{\boldsymbol{W}}_b^T \tilde{\boldsymbol{r}}(k). \tag{25}$$

In general, two strategies including kNN and Bayesian inference are employed to determine the class of new data sample. For kNN, the new data sample is classified by a majority vote of its neighbors, where the class membership is assigned to the class most common among its kNN. For Bayesian inference, the class of new data sample is determined through computation of posterior probability. Compared to kNN algorithm, Bayesian inference strategy is simpler and requires less computational load. In this article, we adopt Bayesian inference for classification. Thus, the Fisher discriminant function $g_j(\tilde{\boldsymbol{r}})$ is utilized to classify the process data

$$\begin{aligned} g_j(\tilde{\boldsymbol{r}}) = &-\frac{1}{2}(\tilde{\boldsymbol{d}}(k) - \tilde{\boldsymbol{d}}_{m,j})^T \left(\frac{1}{n_j - 1}\tilde{\boldsymbol{W}}_b^T \boldsymbol{S}_{w,j} \tilde{\boldsymbol{W}}_b\right)^{-1} \\ &\times (\tilde{\boldsymbol{d}}(k) - \tilde{\boldsymbol{d}}_{m,j}) \\ &- \frac{1}{2}\ln\left[\det\left(\frac{1}{n_j - 1}\tilde{\boldsymbol{W}}_b^T \boldsymbol{S}_{w,j} \tilde{\boldsymbol{W}}_b\right)\right] \end{aligned} \tag{26}$$

where $\tilde{\boldsymbol{d}}_{m,j} = \tilde{\boldsymbol{W}}_b^T \tilde{\boldsymbol{r}}_{m,j}$. Then, the class of the test $\tilde{\boldsymbol{r}}(k)$ is determined by observing the Fisher discriminant function

$$C(\tilde{\boldsymbol{r}}(k)) = \arg\max_{1\leq j\leq c} g_j(\tilde{\boldsymbol{r}}(k)). \tag{27}$$

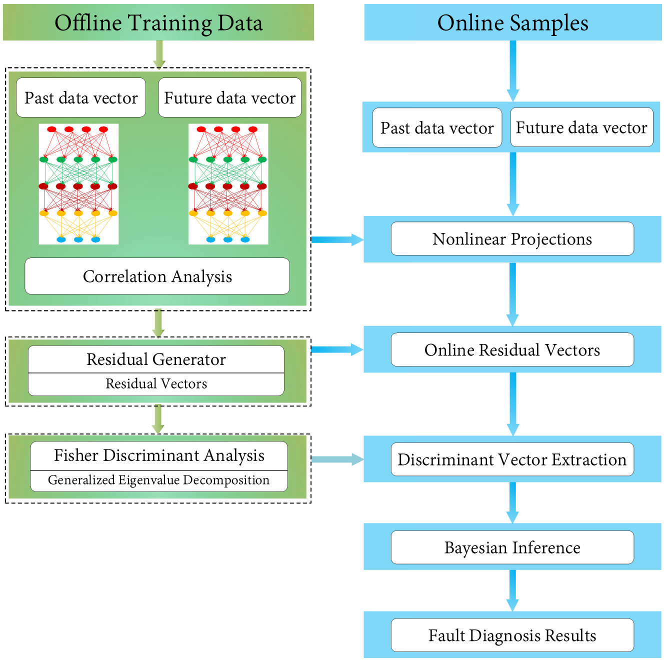


Fig. 1. Scheme of the proposed DCVA-FDA fault diagnosis.

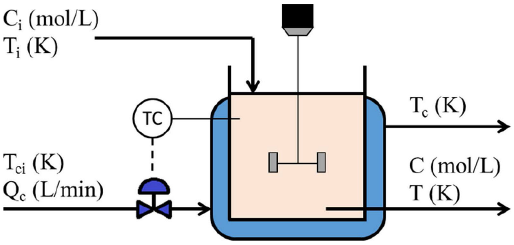


Fig. 2. Flowsheet for the CSTR process [24].

A detailed flowchart for the proposed DCVA-FDA fault diagnosis is shown in Fig. 1.

V. Case Studies

In this section, the proposed DCVA-FDA fault diagnosis scheme is applied to aCSTR and an industrial benchmark of TEP to verify its capability and effectiveness. For comparative study, several typical methods are employed, i.e., FDA, KFDA, KDFDA, and CVA-FDA [17]. To ensure the comparison fairness, all methods use the Bayesian inference based criterion to classify the class label of process data.

A. Case 1: CSTR Process

The flowsheet of CSTR process is shown in Fig. 2. The CSTR process is mainly modeled by the following equations [24]

$$\begin{cases} \frac{dC}{dt} = \frac{Q}{V}(C_i - C) - akC + v_1 \\ \frac{dT}{dt} = \frac{Q}{V}(T_i - T) - a\frac{(\Delta H_r)kC}{\rho C_p} - b\frac{UA}{\rho C_p V}(T - T_c) + v_2 \\ \frac{dT_c}{dt} = \frac{Q_c}{V_c}(T_{ci} - T_c) + b\frac{UA}{\rho_c C_{pe} V_c}(T - T_c) + v_3 \end{cases}$$

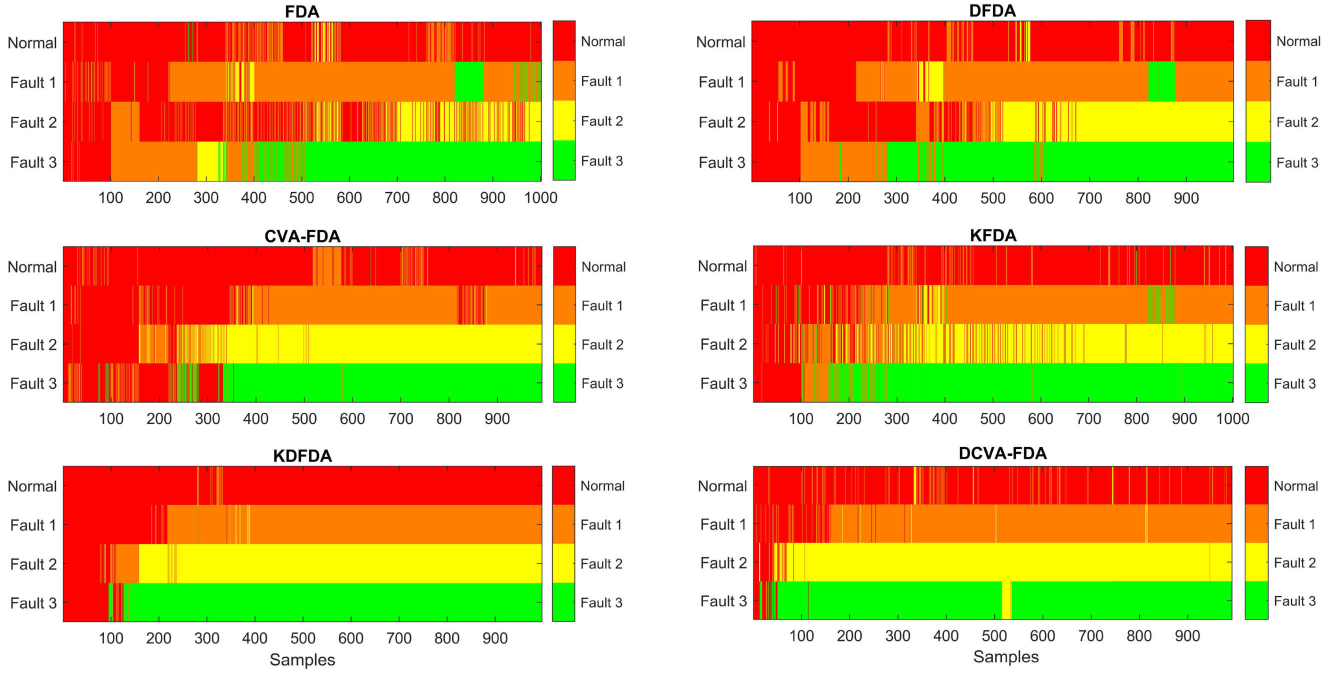


Fig. 3. Classification results for normal and Faults 1–3 testing data in CSTR process. The left axis labels the normal and faulty data and the legend at the right gives the color of the class of process data indicated by the fault diagnosis methods.

TABLE II
DESCRIPTION OF THE FAULTS IN CSTR PROCESS

Fault No.	Description	ξ	Type
1	Sensor drift $T = T + \xi t$	0.005	Additive
2	Catalyst decay $a = a_0 e^{-\xi t}$	5×10^{-4}	Multiplicative
3	Fouling $b = b_0 e^{-\xi t}$	1×10^{-3}	Multiplicative

where C_i is the concentration of the reactant. T_i and T_{ci} are the temperature of reactant and the inlet temperature of coolant, respectively. The symbols $v_i, i = 1, 2, 3$ are process noises. $k = k_0 \exp^{-E/RT}$ is an Arrhenius-type rate constant. Due to the existence of parameter k, the CSTR process becomes dynamic and nonlinear. In this study, we select $\boldsymbol{u} = [C_i \; T_i \; T_{ci}]^T$ and $\boldsymbol{y} = [C \; T \; T_c \; Q_c]^T$. The controller setting and process parameters are referred to [24]. Three typical faults including saturation and sensor drifts are described in Table II.

The sampling interval for all variables is 1 min. 1000 samples are collected in normal and three fault conditions for offline training. Another 1000 samples in normal condition are used for validation in the DCVA-FDA model training. For performance evaluation, a Monte Carlo simulation of five realizations with different random seeds for the process noises, measurement noises, and input disturbances for each condition, is performed to test the consistency of the results. Each testing dataset, which consists of 1000 samples in each condition is generated.

The number of time-lags q is selected as five for CVA-FDA, DFDA, KDFDA, and DCVA-FDA as in [24]. To decide parameters for DCVA-FDA, cross-validation by grid search is employed. The neural networks of DCVA contain five layers (including the output). 100 units per layer are chosen at the input and hidden layers. The dimension of nonlinear representations L is set as 80 for DCVA-FDA. The activation functions at hidden layers are selected as sigmoid function $s(z) = 1/(1 + e^{-z})$, while linear function at input and output layers. The weight decay parameters for all weight parameters are set as 10^{-5}. In this case, the size of minibatch s_{mb} is set as 100. The learning rate ϵ is set as 0.0001. The maximum number of epoch N_{me} is set as 2000. In SGD, the momentum parameter μ is set as 0.99. The regularization parameters λ_p and λ_f are set as 0.01. Additionally, the weight initialization method is according to [37], where Xavier initialization method is applied.

Since there are four classes to be classified, the number of FDA vectors is set as three for all methods. For KFDA and KDFDA, we use a radial basis function kernel. The bandwidth parameters of KFDA and KDFDA are set as 500 and 2500, respectively. These parameters are determined through cross-validation.

The classification results for normal, Faults 1, 2, and 3 utilizing the aforementioned methods are plotted in Fig. 3. As shown in Fig. 3, it can be observed that the normal, Faults 1 and 2 data are often misclassified for linear methods such as FDA, DFDA, and CVA-FDA. In contrast, nonlinear methods such as KFDA, KDFDA, and DCVA-FDA can achieve better classification results than linear methods. Table III shows the average misclassification rate across five testing datasets. As the data listed in Table III, it can further be found that DFDA and CVA-FDA provide better classification results than FDA, as they consider the system dynamics. Moreover, CVA-FDA achieves superior classification performance than DFDA, since the dynamic information can be better captured by pretreating the data with CVA. The overall misclassification rate for CVA-FDA is 23.89%, compared to 40.69% for FDA and 32.71%

TABLE III
AVERAGE MISCLASSIFICATION RATES (%): CSTR PROCESS

Class No.	FDA	DFDA	CVA-FDA	KFDA	KDFDA	DCVA-FDA
Normal	22.3	16.83	10.53	9.27	1.91	9.31
Fault 1	37.94	31.88	30.08	25.57	22.81	10.65
Fault 2	67.93	54.68	27.76	32.41	21.77	6.27
Fault 3	34.61	27.44	27.18	19.34	15.81	7.48
Average	40.69	32.71	23.89	21.65	15.57	8.43

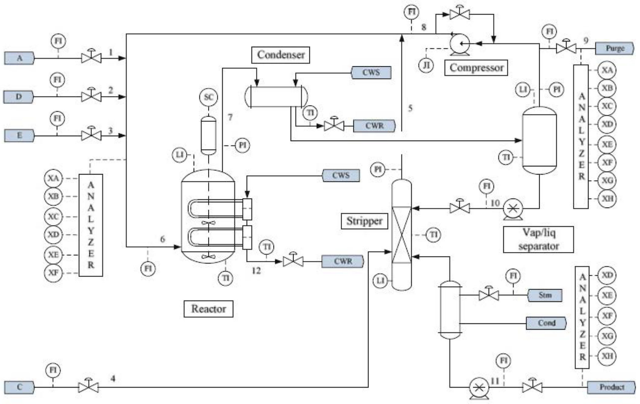


Fig. 4. Diagram of the industrial TEP benchmark [16].

TABLE IV
MONITORING VARIABLES OF THE TEP

	No.	Description	No.	Description
Inputs	XMV(1)	D feed flow	XMV(7)	Separator pot liquid flow
	XMV(2)	E feed flow	XMV(8)	Stripper liquid product flow
	XMV(3)	A feed flow	XMV(9)	Stripper steam valve
	XMV(4)	A and C feed flow	XMV(10)	Reactor cooling water flow
	XMV(5)	Compressor recycle valve	XMV(11)	Condenser cooling water flow
	XMV(6)	Purge valve		
Outputs	XMEAS(1)	A feed	XMEAS(12)	Separator level
	XMEAS(2)	D feed	XMEAS(13)	Separator pressure
	XMEAS(3)	E feed	XMEAS(14)	Separator under flow
	XMEAS(4)	Total Feed (Stream 4)	XMEAS(15)	Stripper level
	XMEAS(5)	Recycle flow	XMEAS(16)	Stripper pressure
	XMEAS(6)	Reactor feed rate	XMEAS(17)	Stripper under flow
	XMEAS(7)	Reactor pressure	XMEAS(18)	Stripper temperature
	XMEAS(8)	Reactor level	XMEAS(19)	Stripper steam flow
	XMEAS(9)	Reactor temperature	XMEAS(20)	Compressor work
	XMEAS(10)	Purge rate	XMEAS(21)	Reactor cooling water outlet temp
	XMEAS(11)	Separator temperature	XMEAS(22)	Condenser cooling water outlet temp

TABLE V
DESCRIPTION OF FAULTS 8, 10, AND 14 IN THE TEP

Fault No.	Description	Type
Fault 8	A, B, C feed composition (stream 4)	Random variation
Fault 10	C feed temperature (stream 4)	Random variation
Fault 14	Reactor cooling water valve	Sticking

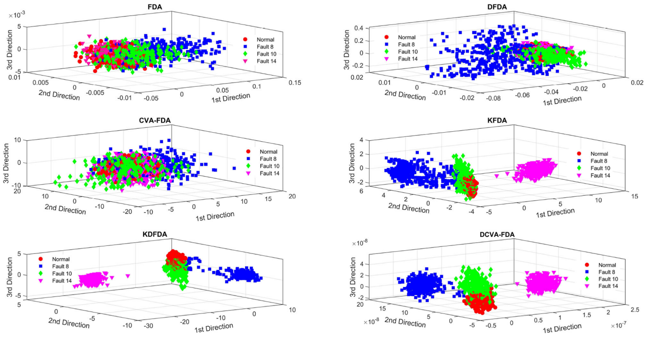


Fig. 5. Label data sample projection results in discriminant subspaces using training data in the TEP.

for DFDA. On the other hand, the nonlinear methods possess better performance than linear methods as displayed in Fig. 3. From the data listed in Table III, the overall misclassification rate for KFDA is 21.65% for the testing datasets. Since KDFDA takes both process dynamics and nonlinearity into account, the overall misclassification rate is 15.57%, which is lower than that of KFDA. Among the nonlinear methods, DCVA-FDA obtains the best classification performance. The combination of DCVA and FDA can enhance the capability of handling process dynamics and nonlinearity. The overall misclassification rate of DCVA-FDA is 8.43%.

B. Case 2: TEP

The industrial benchmark of the TEP was developed based on a practical chemical process [38]. The TEP benchmark has been widely used in the evaluation and comparison of fault diagnosis performance [16]. It consists of five main units including the reactor, condenser, separator, stripper, and compressor. In the TEP, the gaseous reactants A, C, D, E, and the inert B are fed to the reactor where the liquid products G and H are formed. The flowsheet of the TEP is plotted in Fig. 4. There are 11 manipulated variables and 41 measurement variables, among which 22 variables are continuous measurement variables, whereas the other 19 variables are composition measurements. In this article, we adopt 11 manipulated variables XMV(1–11) as inputs and 22 measurement variables XMEAS(1–22) as outputs. Table IV lists the monitoring variables. Total of 21 fault scenarios were simulated in the TEP. More details about the TEP can be found in [16], [38]. Different from the simulation in [16], we choose three different fault scenarios including Faults 8, 10, and 14 for performance evaluation in this work. The descriptions of these fault scenarios are listed in Table V. As shown in Table V, both Faults 8 and 10 are random variations introduced in steam 4, which are composition variation and temperature variation, respectively. Fault 14 is a valve sticking fault from the reactor cooling water valve.

A widely used dataset for fault diagnosis of the TEP can be downloaded from http://web.mit.edu/braatzgroup/links.html. We also adopt this dataset in this work. Based on this dataset, a total of 400 data samples are collected under each of the normal and faulty operating modes for training models. For the purpose of validation in the DCVA-FDA method, another 100 data samples are collected in normal condition. 200 data samples are collected in the normal and faulty conditions for performance evaluation.

For CVA-FDA, DFDA, KDFDA, and DCVA-FDA, the number of time-lags q is selected as five through autocorrelation analysis. Through cross-validation, seven layers (including the output) are used to construct the neural networks in DCVA. The dimensions of $\boldsymbol{z}_p$ and $\boldsymbol{y}_f$ are 165 and 110, respectively. At the input and hidden layers, 200 units per layer are chosen. The dimension of projections L is set as 180 for DCVA-FDA. At the hidden layer, the rectified linear unit (ReLU) is adopted as activation function. The expression of ReLU is $s(z) = \max(0, z)$.

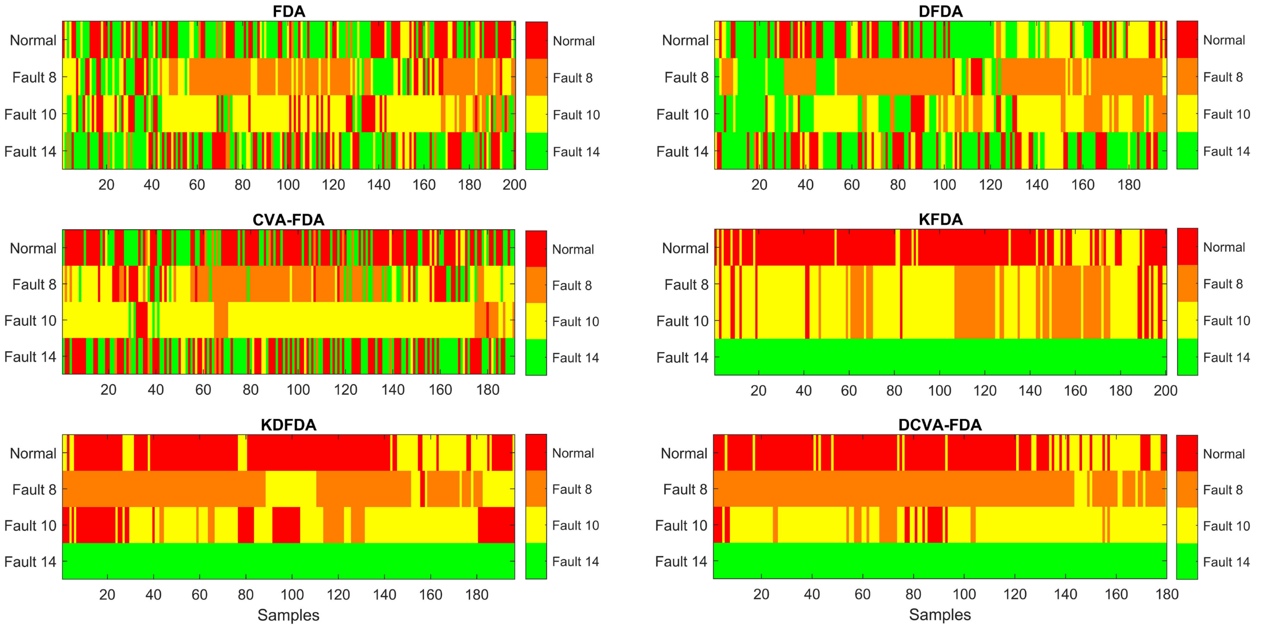


Fig. 6. Classification results for normal and Faults 8, 10, and 14 testing data in the TEP. The left axis labels the normal and faulty data and the legend at the right gives the color of the class of process data indicated by the fault diagnosis methods.

The linear function is chosen at input and output layers. The weight decay parameters for all weight parameters are set as 10^{-5}. The size of minibatch s_{mb} is set as 300. The learning rate ϵ is set as 0.01. For SGD, the momentum parameter μ is set as 0.7. The regularization parameters λ_p and λ_f are set as 0.0001. Different from sigmoid function, the weight initialization is using random initialization method while ReLU is adopted, according to [39]. The maximum number of epoch N_{me} is set as 2000. Three FDA vectors are chosen as there are four classes of process data to classify for all methods. For KFDA and KDFDA, the radial basis function kernel is used. The bandwidth parameters of KFDA and KDFDA are set as 1650 and 10 000, respectively. These parameters are determined through cross-validation.

To illustrate the discriminant capability of all methods, all the labeled training data samples are projected into the subspaces created by the discriminant vectors. In Fig. 5, the first three directions of the projections are plotted. From Fig. 5, it can be seen that it is very difficult to separate samples from both normal and faulty scenarios in the discriminant subspace provided by the linear models such as FDA, DFDA, and CVA-FDA. In contrast, the labeled sample projections are better separated by nonlinear models such as KFDA, KDFDA, and DCVA-FDA. For nonlinear methods, although the distance between the projections of normal and Fault 10 data is closer, the projections of Faults 8 and 14 are far away from the centers of the projections of normal and Fault 10 data. It indicates that the process data, which are generated from Faults 8 and 14 scenarios can be classified by nonlinear methods more accurately than linear methods. Experimental results regarding the testing dataset are presented in Fig. 6. As shown in Fig. 6, the classification performance is improved using CVA-FDA compared to FDA and DFDA, since more information on dynamics is revealed from the process data. In Table VI, the misclassification rates are listed. The overall misclassification rate for CVA-FDA is 42.15% for the testing dataset, compared to 51.00% for FDA and 51.53% for DFDA. Due to the strong nonlinear characteristic of the TEP, the nonlinear methods are superior over linear methods as shown in Fig. 6. The overall misclassification rate for KFDA is 32.00%. Both the dynamics and nonlinearity of process data are taken into consideration by KDFDA. Thus, the classification performance utilizing KDFDA is better than KFDA. The overall misclassification rate of KDFDA is 22.70%. DCVA facilitates the extraction of discriminant features from process data. As we expected, DCVA-FDA can achieve the best classification performance among all the compared methods. The overall misclassification rate of DCVA-FDA is 12.30%. Thus, it can be concluded that the combination of DCVA and FDA is advantageous for fault diagnosis compared with other methods.

TABLE VI
Misclassification Rates (%): TEP

Class No.	FDA	DFDA	CVA-FDA	KFDA	KDFDA	DCVA-FDA
Normal	60.50	71.94	40.84	20.50	26.02	23.56
Fault 8	52.00	30.61	58.64	69.50	22.96	6.81
Fault 10	40.00	55.10	14.14	38.00	41.84	18.85
Fault 14	51.50	48.47	54.97	0.00	0.00	0.00
Average	51.00	51.53	42.15	32.00	22.70	12.30

VI. Conclusion

In this article, a novel data-driven fault diagnosis method named DCVA-FDA was proposed for nonlinear dynamic processes. A new nonlinear CVA was developed by incorporating DNN into CVA to learn the nonlinear dynamic relationship between process variables from data. A residual generator was designed from the DCVA model. FDA was performed on the residual vectors for discriminant analysis. Then, Bayesian inference was applied to classify the process data. The proposed DCVA-FDA took both advantages of DCVA and FDA for fault diagnosis. Two experimental studies were carried out to validate the superiority of the proposed DCVA-FDA scheme by comparing it with reported fault diagnosis methods. Although DCVA-FDA could achieve improved performance, the misclassification rate of normal conditions was relatively high. In this article, the most commonly used DNN architecture was employed in DCVA-FDA. Hence, future investigations would consider more DNN architectures to enhance the ability of nonlinear representation. Moreover, given the advantages of DCVA in learning deep correlated representations for nonlinear dynamic processes, the proposed DCVA-FDA method would be applied in more research areas such as quality-related fault diagnosis, semisupervised fault diagnosis in future work.

Acknowledgment

The authors would like to thank the anonymous reviewers for their comments and suggestions.

References

[1] Z. Gao, C. Cecati, and S. X. Ding, “A survey of fault diagnosis and fault-tolerant techniques-part I: Fault diagnosis with model-based and signal-based approaches,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3757–3767, Jun. 2015.

[2] R. Isermann, *Fault-Diagnosis Systems*. Berlin, Germany: Springer, 2006.

[3] K. Tidriri, N. Chatti, S. Verron, and T. Tiplica, “Bridging data-driven and model-based approaches for process fault diagnosis and health monitoring: A review of researches and future challenges,” *Annu. Rev. Control*, vol. 42, pp. 63–81, 2016.

[4] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, “A review of process fault detection and diagnosis: Part I: Quantitative model-based methods,” *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 293–311, 2003.

[5] V. Venkatasubramanian, R. Rengaswamy, and S. N. Kavuri, “A review of process fault detection and diagnosis: Part II: Qualitative models and search strategies,” *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 313–326, 2003.

[6] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin, “A review of process fault detection and diagnosis: Part III: Process history based methods,” *Comput. Chem. Eng.*, vol. 27, no. 3, pp. 327–346, 2003.

[7] Z. Ge, Z. Song, S. X. Ding, and B. Huang, “Data mining and analytics in the process industry: The role of machine learning,” *IEEE Access*, vol. 5, pp. 20 590–20 616, 2017.

[8] S. Ding, P. Zhang, T. Jeinsch, E. Ding, P. Engel, and W. Gui, “A survey of the application of basic data-driven and model-based methods in process monitoring and fault diagnosis,” *IFAC Proc. Vol.*, vol. 44, no. 1, pp. 12 380–12 388, 2011.

[9] S. Yin, S. X. Ding, X. Xie, and H. Luo, “A review on basic data-driven approaches for industrial process monitoring,” *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6418–6428, Nov. 2014.

[10] Y. Tao, H. Shi, B. Song, and S. Tan, “A novel dynamic weight principal component analysis method and hierarchical monitoring strategy for process fault detection and diagnosis,” *IEEE Trans. Ind. Electron.*, vol. 67, no. 9, pp. 7994–8004, Sep. 2020.

[11] R. Muradore and P. Fiorini, “A PLS-based statistical approach for fault detection and isolation of robotic manipulators,” *IEEE Trans. Ind. Electron.*, vol. 59, no. 8, pp. 3167–3175, Aug. 2012.

[12] P. P. Odiowei and Y. Cao, “Nonlinear dynamic process monitoring using canonical variate analysis and kernel density estimations,” *IEEE Trans. Ind. Informat.*, vol. 6, no. 1, pp. 36–45, Feb. 2010.

[13] 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.

[14] S. J. Qin, “Data-driven fault detection and diagnosis for complex industrial processes,” *IFAC Proc. Vol.*, vol. 42, no. 8, pp. 1115–1125, 2009.

[15] S. J. Qin and L. H. Chiang, “Advances and opportunities in machine learning for process data analytics,” *Comput. Chem. Eng.*, vol. 126, pp. 465–473, 2019.

[16] R. B. L.H. Chiang and E.L. Russell, *Fault Detection and Diagnosis in Industrial Systems*. London, U.K.: Springer, 2001.

[17] B. Jiang, X. Zhu, D. Huang, J. A. Paulson, and R. D. Braatz, “A combined canonical variate analysis and fisher discriminant analysis (CVA-FDA) approach for fault diagnosis,” *Comput. Chem. Eng.*, vol. 77, pp. 1–9, 2015.

[18] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K. R. Mullers, “Fisher discriminant analysis with kernels,” in *Proc. Neural Netw. Signal Process. IX Proc. IEEE Signal Process. Soc. Workshop*, 1999, pp. 41–48.

[19] Z. Ge, S. Zhong, and Y. Zhang, “Semisupervised kernel learning for FDA model and its application for fault classification in industrial processes,” *IEEE Trans. Ind. Informat.*, vol. 12, no. 4, pp. 1403–1411, Aug. 2016.

[20] J. Feng, J. Wang, H. Zhang, and Z. Han, “Fault diagnosis method of joint fisher discriminant analysis based on the local and global manifold learning and its kernel version,” *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 1, pp. 122–133, Jan. 2016.

[21] C. Sumana, B. Mani, C. Venkateswarlu, and R. D. Gudi, “Improved fault diagnosis using dynamic kernel scatter-difference-based discriminant analysis,” *Ind. Eng. Chem. Res.*, vol. 49, no. 18, pp. 8575–8586, 2010.

[22] Q. Jiang and X. Yan, “Learning deep correlated representations for nonlinear process monitoring,” *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6200–6209, Dec. 2019.

[23] Y. Liu, B. Liu, X. Zhao, and M. Xie, “A mixture of variational canonical correlation analysis for nonlinear and quality-relevant process monitoring,” *IEEE Trans. Ind. Electron.*, vol. 65, no. 8, pp. 6478–6486, Aug. 2018.

[24] K. E. S. Pilario and Y. Cao, “Canonical variate dissimilarity analysis for process incipient fault detection,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 12, pp. 5308–5315, Dec. 2018.

[25] Z. Chen, S. X. Ding, T. Peng, C. Yang, and W. Gui, “Fault detection for non-gaussian processes using generalized canonical correlation analysis and randomized algorithms,” *IEEE Trans. Ind. Electron.*, vol. 65, no. 2, pp. 1559–1567, Feb. 2018.

[26] Z. Chen, S. X. Ding, K. Zhang, Z. Li, and Z. Hu, “Canonical correlation analysis-based fault detection methods with application to alumina evaporation process,” *Control Eng. Pract.*, vol. 46, pp. 51–58, 2016.

[27] S. Zhang, C. Zhao, and B. Huang, “Simultaneous static and dynamic analysis for fine-scale identification of process operation statuses,” *IEEE Trans. Ind. Informat.*, vol. 15, no. 9, pp. 5320–5329, Sep. 2019.

[28] R. Salakhutdinov and G. Hinton, “Deep Boltzmann machines,” in *Proc. 12th Int. Conf. Artif. Intell. Statist.*, Apr. 2009, vol. 5, pp. 448–455.

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

[30] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A fast learning algorithm for deep belief nets,” *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[31] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, “Deep canonical correlation analysis,” in *Proc. 30th Int. Conf. Mach. Learn.*, Jun. 2013, vol. 28, no. 3, pp. 1247–1255.

[32] Y. Yu, S. Tang, K. Aizawa, and A. Aizawa, “Category-based deep CCA for fine-grained venue discovery from multimodal data,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 4, pp. 1250–1258, Apr. 2019.

[33] W. Wang, R. Arora, K. Livescu, and J. A. Bilmes, “Unsupervised learning of acoustic features via deep canonical correlation analysis,” in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2015, pp. 4590–4594.

[34] Hanh Vu, B. Koo, and S. Choi, “Frequency detection for SSVEP-based BCI using deep canonical correlation analysis,” in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2016, pp. 1983–1987.

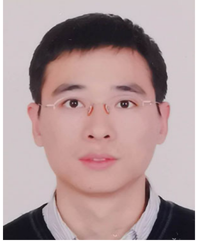
[35] B. Jiang and R. D. Braatz, “Fault detection of process correlation structure using canonical variate analysis-based correlation features,” *J. Process. Control*, vol. 58, pp. 131–138, 2017.

[36] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[37] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, vol. 9, pp. 249–256.

[38] J. Downs and E. Vogel, “A plant-wide industrial process control problem,” *Comput. Chem. Eng.*, vol. 17, no. 3, pp. 245–255, 1993.

[39] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proc. IEEE Int. Conf. Comput. Vis.,* 2015, pp. 1026–1034.



Ping Wu received the B.S. and Ph.D. degrees in control theory and control engineering from Zhejiang University, Hangzhou, China, in 2003 and 2009, respectively.

He is currently an Associate Professor with the Faculty of Mechanical Engineering and Automation, Zhejiang Sci-Tech University. His research interests include fault diagnosis, machine learning, and industrial intelligence.

Siwei Lou received the B.S. degree in automation in 2018 from Zhejiang Sci-Tech University, Hangzhou, China, where he is currently working toward the M.S. degree in control science and engineering from the Faculty of Mechanical Engineering and Automation.

His current research interests include process monitoring, deep learning, and fault diagnosis.

Xujie Zhang received the B.S. degree in mechanical engineering in 2019 from Zhejiang Sci-Tech University, Hangzhou, China, where he is currently working toward the M.S. degree in control science and engineering from the Faculty of Mechanical Engineering and Automation.

His current research interests include deep learning and fault diagnosis.

Jiajun He received the B.S. degree in automation in 2019 from Zhejiang Sci-Tech University, Hangzhou, China, where he is currently working toward the M.S. degree in control science and engineering from the Faculty of Mechanical Engineering and Automation.

His current research interests include deep learning and fault diagnosis.

Yichao Liu received the B.S. degree (with honors) in marine technology from Dalian Maritime University, Dalian, China, in 2014 and the Ph.D. degree in environmental science and engineering from Tsinghua University, Beijing, China, in 2018.

He is currently a Postdoctoral Researcher and Marie-Curie Fellow with the Delft University of Technology, Delft, The Netherlands, in the Delft Center for Systems and Control. His research interests include dynamical modeling, wind turbine control and fault-tolerant control.

Jinfeng Gao received the B.S. degree from the Hebei Institute of Science and Technology, Hebei, China, in 2000, the M.S. degree from the Zhejiang University of Technology, Hangzhou, China, in 2003, and the Ph.D. degree from Zhejiang University, Hangzhou, China, in 2008, all in control engineering.

She is currently a Professor with Zhejiang Sci-Tech University, Hangzhou. Her research interests include fault detection and diagnosis, networked control, and multiagent systems.