processes

MDPI

*Article*

# Gap-MK-DCCA-Based Intelligent Fault Diagnosis for Nonlinear Dynamic Systems

**Junzhou Wu, Mei Zhang * and Lingxiao Chen**

The Electrical Engineering College, Guizhou University, Guiyang 550025, China; gs.jzwu21@gzu.edu.cn (J.W.); gs.xlchen22@gzu.edu.cn (L.C.)
* Correspondence: mzhang3@gzu.edu.cn; Tel.: +86-13385104650

**Abstract:** In intelligent process monitoring and fault detection of the modern process industry, conventional methods mostly consider singular characteristics of systems. To tackle the problem of suboptimal incipient fault detection in nonlinear dynamic systems with non-Gaussian distributed data, this paper proposes a methodology named Gap-Mixed Kernel-Dynamic Canonical Correlation Analysis. Initially, the Gap metric is employed for data preprocessing, followed by fault detection utilizing the Mixed Kernel-Dynamic Canonical Correlation Analysis. Ultimately, fault identification is conducted through a contribution method based on the $T^2$ statistic. Furthermore, a comparative analysis was conducted using Canonical Variate Analysis, Dynamic Canonical Correlation Analysis, and Mixed Kernel-Dynamic Canonical Correlation Analysis on the Tennessee Eastman process. Experimental results indicate varying degrees of improvements in the detection rate, false alarm rate, missed detection rate, and detection time compared to the comparative methods, demonstrating the industrial value and academic significance of the method.

**Keywords:** gap metric; canonical correlation analysis; kernel density estimate; fault detection; Tennessee Eastman process



**Citation:** Wu, J.; Zhang, M.; Chen, L. Gap-MK-DCCA-Based Intelligent Fault Diagnosis for Nonlinear Dynamic Systems. *Processes* **2024**, *12*, 388. https://doi.org/10.3390/pr12020388

Academic Editor: Stefania Tronci

Received: 17 January 2024
Revised: 7 February 2024
Accepted: 8 February 2024
Published: 15 February 2024



## 1. Introduction

Modern process industries are undergoing a transformation from traditional to efficient, large-scale, complex, and integrated systems, adapting to escalating market demands for a wide variety of high-value-added products with multiple specifications. Within this developmental trend, process industrial systems in sectors like steel, pharmaceuticals, and petrochemicals have evolved into highly automated and complex large-scale production systems. Nevertheless, with the expansion of scale, the high-intensity operation of systems may result in minor faults or deviations from the optimal working conditions in any unit or subsystem. If these faults are not promptly detected and addressed, they could propagate, worsen, or even trigger chain reactions, causing impacts ranging from compromised product quality to severe consequences such as equipment damage, environmental pollution, substantial property losses, and even casualties or fatalities in major safety incidents. Historical incidents such as the gas leak at the Carbide Company in India and the nuclear leakage at the Fukushima Nuclear Power Plant serve as warnings, indicating that safe, reliable, high-quality, and efficient operation is crucial in modern process industries. Consequently, providing effective safeguarding for the long-term operation of process industries through process monitoring technology has emerged as a crucial component of sustainable development in process industries. It carries strategic significance in enhancing product quality and ensuring the secure operation of processes, marking it as a current research focus in the field of process industry control with significant theoretical value and extensive prospects for engineering applications.

However, the field of intelligent process monitoring and fault detection in process industries is encountering challenges posed by the complexity and high integration of

systems. Traditional process monitoring methods appear somewhat inadequate in dealing with the dynamics, non-linearity, and complexity of process industrial systems. The mutual coupling of numerous production processes and intricate flow patterns of fluid, energy, and information render traditional process monitoring methods insufficient in adapting to the complexity of process industries. The specific characteristics of process industries, including nonlinearity, time variation, and multimodality, pose additional challenges for traditional monitoring methods.

The majority of traditional process monitoring methods are rooted in multivariate statistical analysis techniques, including Principal Component Analysis (PCA), Partial Least Squares (PLS), Canonical Variate Analysis (CVA), Canonical Variate Dissimilarity Analysis (CVDA), and Canonical Correlation Analysis (CCA), which have been extensively researched by a wide range of scholars and have yielded noteworthy research achievements and a plethora of successful applications [1]. For instance, references [2–7] have, respectively, investigated and improved the PCA method to varying extents and achieved successful applications in the industry; Ding et al. applied an enhanced PLS to predict and diagnose key performance indicators of industrial hot-rolled strip steel mills [8], and a series of researches were similarly carried out for the PLS method in [9–13]; Ruiz-Cárcel and colleagues achieved satisfactory experimental results in the process monitoring of multiphase flow facilities using the CVA method [14]; subsequently, Pilario et al. proposed the CVDA method and its extended version based on CVA, offering new insights into incipient fault detection in dynamic systems [15–17]; as for the CCA method, Chen et al. pioneered the use of data-driven CCA technology for generating residuals based on canonical correlation, applying it to fault detection in both static and dynamic processes [18]; following this, the CCA method has gradually attracted attention from scholars in the field of process monitoring and fault detection, undergoing extensive research and improvement [19–28].

Nevertheless, traditional CCA methods are unable to meet the requirement of dynamics in modern process industry systems for process monitoring. Hence, Chen et al. extended it to a dynamic version, proposing the DCCA method [18]. Additionally, the high nonlinearity and non-Gaussian distribution of data in dynamic systems pose significant challenges to process monitoring and fault detection in process industries. Although Pilario et al. proposed the Mixed Kernel- Canonical Variate Dissimilarity Analysis (MK-CVDA) method, utilizing a mixed kernel to address system nonlinearity [17], its performance in the Tennessee Eastman Process (TEP) was somewhat inadequate. It failed to investigate dynamic relationships between system inputs and outputs and did not resolve the issue of non-Gaussian data distribution. To address these concerns, we introduced the Mixed Kernel-Dynamic Canonical Correlation Analysis (MK-DCCA) method in our previous work, achieving effective handling of system dynamics and nonlinearity in a simulation model of the Continuous Stirred Tank Reactor (CSTR) with positive experimental results [29]. However, the MK-DCCA method still showed unsatisfactory performance in TEP, largely due to the imperfect handling of system nonlinearity by the mixed kernel method and the unresolved issue of non-Gaussian data distribution.

To tackle the aforementioned challenges, we propose a novel intelligent fault diagnosis method based on the Gap metric, MK-DCCA, and Kernel Density Estimation (KDE), named Gap-Mixed Kernel-Dynamic Canonical Correlation Analysis (Gap-MK-DCCA), with the objective of addressing fault detection issues in modern process industry systems attributed to their dynamics, nonlinearity, and non-Gaussian data distribution characteristics. In this approach, building on MK-DCCA, the Gap metric is introduced as a data preprocessing step, avoiding the problem of inaccurate feature extraction caused by traditional preprocessing methods. This enhancement allows for a better representation of the correlation among system variables in high-dimensional space, effectively avoiding high coupling between variables and further optimizing the fault detection capability of the method in nonlinear systems. Moreover, to address the issue of non-Gaussian data distribution, Kernel Density Estimation is introduced for threshold calculation. Compared to traditional methods, Gap-MK-DCCA exhibits superior capabilities in handling dynamics, nonlinearity, and non-

Gaussian characteristics, making it an advanced solution for intelligent process monitoring and fault detection in modern process industries.

The rest of this paper is organized as follows: in Section 2, a brief exposition of the fundamental theories related to this method is provided; in Section 3, the Gap-MK-DCCA method used in this study is introduced and detailed; in Section 4, a case analysis of the Gap-MK-DCCA method is conducted, comparing it with CVA, DCCA, and MK-DCCA methods through experiments to validate the superior performance of the proposed method; in Section 5, a summary of the work in this paper is provided.

## 2. Preliminary Work

In previous work, we demonstrated the effectiveness of the MK-DCCA method for diagnosing incipient faults in nonlinear dynamic systems through process monitoring and fault detection using a simulation model of CSTR [29]. Building upon the foundation of the previous research, this study aims to further optimize the method to enhance its diagnostic performance in the face of various complex characteristics of incipient fault diagnosis in dynamic systems. To accomplish this, we will begin with a brief review of the method.

### 2.1. Mixed Kernel Method

Kernel methods, widely acknowledged for their effectiveness in handling nonlinearity, often have their generalization performance influenced by the choice of kernel function. To bestow the kernel function with proficiency in both interpolation and extrapolation, namely excellent generalization performance, we choose to employ a combination scheme based on Kernel Principal Component Analysis (KPCA). This scheme involves combining the Radial Basis Function (RBF) kernel and the polynomial kernel into a mixed kernel. Here is a detailed presentation of the related formulas:

$$K_{RBF}(x, x^{'}) = \exp(-\frac{\left\| x - x^{'} \right\|^2}{c}) \tag{1}$$

$$K_{poly}(x, x^{'}) = ((x, x^{'}) + 1)^d \tag{2}$$

$$K_{mix} = \omega K_{poly} + (1 - \omega) K_{RBF} \tag{3}$$

where $K_{RBF}$ represents the Gaussian kernel function, $(x, x^{'})$ denotes a scalar product, and $c$ is the kernel width, satisfying the Mercer condition for $c > 0$ [30]. $K_{poly}$ is the polynomial kernel function, $d$ is the kernel parameter representing the polynomial degree, satisfying the Mercer condition for $d \in \mathbb{N}$ [31]. $K_{mix}$ denotes the mixed kernel function, $\omega \in [0, 1]$ is the mixing weight, the mixed kernel transitions into the polynomial kernel and the RBF kernel at $\omega = 1$ and $\omega = 0$, respectively.

Jordaan et al. demonstrated that the weighted sum of the linear polynomial kernel ($d = 1$) and the RBF kernel can optimize the generalization performance of kernel methods [32]. Therefore, the polynomial kernel with $d = 1$ and the RBF kernel are chosen for combination. Furthermore, in our previous work, through the grid search method, it was demonstrated that $c = 4.5$ is the optimal kernel width for RBF, with the optimal weight being $\omega = 0.95$ [29].

Once the kernel function and its parameters are determined, applying the KPCA method to the sampled data enables the transformation of a non-linear data matrix into a linear data matrix.

### 2.2. Dynamic Canonical Correlation Analysis Method

Due to the inability of traditional CCA methods to meet the requirements of modern process industries for system dynamics, Chen et al. proposed extending it to the dynamic version, DCCA, by introducing lag and lead parameters [18]. Assuming $p$ and $f$ are lag

and lead parameters, respectively, and $u \in \mathbb{R}^l$ and $y \in \mathbb{R}^m$ are input and output vectors, we initially define the data matrix and set as follows:

$$u_p(i) = \begin{bmatrix} u(i-p) \\ \vdots \\ u(i-1) \end{bmatrix} \tag{4}$$

$$y_p(i) = \begin{bmatrix} y(i-p) \\ \vdots \\ y(i-1) \end{bmatrix} \tag{5}$$

$$u_f(i) = \begin{bmatrix} u(i) \\ \vdots \\ u(i+f) \end{bmatrix} \tag{6}$$

$$y_f(i) = \begin{bmatrix} y(i) \\ \vdots \\ y(i+f) \end{bmatrix} \tag{7}$$

Here, $u_p(i)$ and $y_p(i)$ represent the past input matrix and past output matrix, while $u_f(i)$ and $y_f(i)$ denote the future input matrix and future output matrix, respectively. The new input and output matrices are constructed using the following formulas:

$$z_p(i) = \begin{bmatrix} y_p(i) \\ u_p(i) \end{bmatrix} \tag{8}$$

$$Z_p = [z_p(1), \cdots, z_p(N)] \in \mathbb{R}^{(p(m+l)\times N)} \tag{9}$$

$$U_f = [u_f(1), \cdots, u_f(N)] \in \mathbb{R}^{(f+1)l\times N)} \tag{10}$$

$$Y_f = [y_f(1), \cdots, y_f(N)] \in \mathbb{R}^{(f+1)m\times N)} \tag{11}$$

$$\begin{aligned} Z &= \begin{bmatrix} Z_p \\ U_f \end{bmatrix}; \\ Y &= Y_f \end{aligned} \tag{12}$$

where $N$ represents the number of samples, and $Z$ and $Y$ are the new input and output matrices, respectively. Subsequently, by applying the CCA method to $Z$ and $Y$, the monitoring statistic indicator $T^2$ is calculated, and the formula for the corresponding threshold is as follows:

$$T^2_{UCL} = \frac{k(N^2-k)}{N(N-k)} \mathcal{F}_{1-\alpha}(k, N-k) \tag{13}$$

Here, $\alpha$ represents the confidence level, $k$ is the number of retained singular values chosen, and $N$ is the number of samples. After acquiring the threshold value, the procedure of process monitoring is carried out in accordance with the subsequent logic: $T^2 > T^2_{UCL}$ indicates a fault; otherwise, the system is considered normal.

## 3. Gap-Mixed Kernel-Dynamic Canonical Correlation Analysis Method

In this section, we extensively present the Gap-Mixed Kernel-Dynamic Canonical Correlation Analysis (Gap-MK-DCCA) method proposed in this paper, which mainly includes the data preprocessing method based on the Gap metric, the fault detection method based on MK-DCCA, and the fault identification method based on the contribution of the $T^2$ statistic. Figure 1 provides a detailed description of these three parts of the Gap-MK-DCCA method.

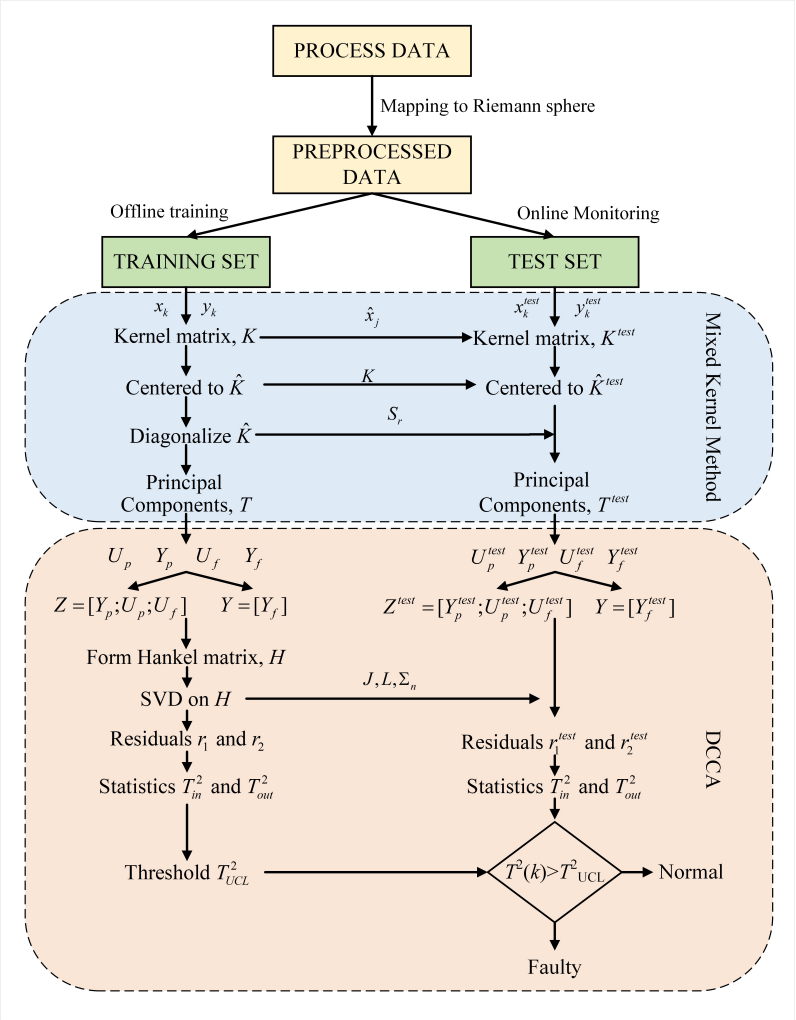


**Figure 1.** Detailed flowchart of the Gap-MK-DCCA approach.

### *3.1. Gap-Metric-Based Data Preprocessing*

Traditional data preprocessing methods tend to overlook the correlation between different variables in Euclidean space, resulting in inaccurate feature extraction. However, the Gap-metric-based data preprocessing method provides a better reflection of the correlation among various system variables in high-dimensional space while effectively avoiding high coupling situations between variables and improving issues related to nonlinear data to different degrees. Therefore, in this study, we choose the Gap-metric-based data preprocessing method as a replacement for traditional z-score and max–min preprocessing methods.

Initially, we assume the data matrix $X_n \in \mathbb{R}^{N\times n}$ for the multidimensional variable system is as follows.

$$X_n = \begin{bmatrix} x_1(1) & x_1(2) & \cdots & x_1(n) \\ x_2(1) & x_2(2) & \cdots & x_2(n) \\ \vdots & \vdots & \ddots & \vdots \\ x_N(1) & x_N(2) & \cdots & x_N(n) \end{bmatrix} \tag{14}$$

where $n$ represents the number of system variables, and $N$ represents the number of samples. Subsequently, the matrix $X_n$ is subjected to mean normalization using the following procedure:

$$b_n = \frac{1}{N} l_N X_n \tag{15}$$

Here, $l_N = [1, 1, \cdots, 1] \in \mathbb{R}^{1\times N}$, and $b_n$ denotes the sample mean vector for each variable in $X_n$. Subsequently, raw data are projected onto the Riemannian sphere, and the Gap metric of the sample distance of each variable from its sample mean is calculated. The transformed matrix is denoted as $X^*$, and the relevant equation is shown below:

$$X^* = \begin{bmatrix} \delta(x_1(1), b_n(1)) & \delta(x_1(2), b_n(2)) & \cdots & \delta(x_1(n), b_n(n)) \\ \delta(x_2(1), b_n(1)) & \delta(x_2(2), b_n(2)) & \cdots & \delta(x_2(n), b_n(n)) \\ \vdots & \vdots & \ddots & \vdots \\ \delta(x_N(1), b_n(1)) & \delta(x_N(2), b_n(2)) & \cdots & \delta(x_N(n), b_n(n)) \end{bmatrix} \tag{16}$$

where

$$\delta(x_i(j), b_n(j)) = \|\varphi(x_i(j) - \varphi(b_n(j))\| = \frac{|x_i(j) - b_n(j)|}{\sqrt{1 + (x_i(j))^2}\sqrt{1 + (b_n(j))^2}} \tag{17}$$

### *3.2. Process Monitoring and Fault Detection Based on the MK-DCCA Method*

#### 3.2.1. MK-DCCA Methodology

The process monitoring and fault detection using preprocessed data matrix $X^*$ are primarily divided into two stages: offline training and online monitoring. In the offline training stage, the preprocessed data matrices are firstly divided into input matrix $Z_{train}$ and output matrix $Z_{train}$. Then, the self-covariance matrix and cross-covariance matrix are calculated for each and used to generate the Hankel matrix $H$. Following the singular value decomposition (SVD) of the matrix $H$ to obtain left and right singular vectors $u$ and $v$ and singular value matrix $s$, the projection matrices $J$ and $L$, maximizing correlation between input and output matrices, can be computed using the following formulas:

$$H = \Sigma_{ZZ}^{-1/2}\Sigma_{ZY}\Sigma_{YY}^{-1/2} \tag{18}$$

$$J = \Sigma_{ZZ}^{-1/2}u(:, 1:k) \tag{19}$$

$$L = \Sigma_{YY}^{-1/2}v(:, 1:k) \tag{20}$$

Here, $\Sigma_{ZZ}$ and $\Sigma_{YY}$ are self-covariance matrices of input and output matrices, $\Sigma_{ZY}$ is a cross-covariance matrix between input and output matrices, and $k$ represents the number of retained singular values; its value can be determined using the Cumulative Percentage Value (CPV) method [33]. Once projection matrices $J$ and $L$ are obtained, they are transferred for backup to the online monitoring stage. Simultaneously, they are employed for calculating residual $r$ and monitoring statistic $T^2$:

$$\begin{aligned} r_1(i) &= J^T z(i) - sL^T y(i) \\ r_2(i) &= L^T y(i) - s^T J^T z(i) \end{aligned} \tag{21}$$

$$T^2(i) = (N-1)r^T(i)(I - s_k^2)^{-1}r(i) \tag{22}$$

where $i$ represents the data at the $i$-th sampling instant, $s$ denotes the singular value matrix, $I$ is the corresponding identity matrix, and $k$ is the number of retained singular values. Based on the two residuals $r_1(i)$ and $r_2(i)$, Equation (24) can be used to calculate the two statistics $T_{in}^2$ and $T_{out}^2$, which are employed for achieving optimal fault detection for faults occurring in input subspace and in output subspace, respectively.

Ultimately, threshold calculation based on the statistic $T^2$ using the KDE method will be elaborated on in Section 3.2.2.

In the online monitoring stage, preprocessed data matrices are similarly partitioned to obtain the input matrix $Z_{test}$ and the output matrix $Y_{test}$. Afterward, utilizing projection matrices $J$ and $L$ obtained in the offline training stage through Formulas (23) and (24), we calculate the residual $r$ and the statistic $T^2$ during the online monitoring stage. Fault

detection is then carried out following the logic below: $T^2 > T^2_{UCL}$ indicates a fault, and further fault identification is performed for accurate fault localization; otherwise, the system is considered normal.

#### 3.2.2. KDE-Based Threshold Calculation

Most traditional threshold calculation methods assume that data follow a Gaussian distribution. However, processes in modern process industrial systems are generally non-Gaussian, which renders the thresholds calculated from Equation (13) unsuitable for real-time monitoring. To address this issue, we choose to directly estimate the Probability Density Function (PDF) of the statistic through non-parametric methods [34,35].

Among various PDF estimation methods, the Kernel Density Estimation (KDE) method exhibits stronger adaptability and flexibility in handling complex, multi-modal, high-dimensional data and anomaly detection. Hence, the KDE method is chosen for the threshold calculation in this paper. Suppose $x$ is a random variable with a density function denoted by $p(x)$, which means

$$P(x < b) = \int_{-\infty}^{b} p(x)dx \tag{23}$$

Thus, as long as $p(x)$ is known, (25) can be utilized to determine appropriate control limits for a specific confidence interval $\alpha$. The estimation of the probability density function $\hat{p}(x)$ at point $x$ through the kernel function $K(\cdot)$ is defined as follows:

$$\hat{p}(x) = \frac{1}{Mh} \sum_{k=1}^{M} K(\frac{x - x_k}{h}) \tag{24}$$

where $x_k, k = 1, 2, \cdots, M$ is a sample from $x$, $h$ is the bandwidth, and a radial basis function is chosen, expressed as follows:

$$K(x) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{g^2}{2}) \tag{25}$$

By replacing $x_k$ with the statistic $T^2_k$, the above KDE method can estimate the PDF of the indicator $T^2$. Subsequently, given a confidence level $\alpha$, the corresponding control limits can be obtained from the PDF of the indicator by solving the following equation:

$$\int_{-\infty}^{T^2_{UCL}(\alpha)} p(T^2)dT^2 = \alpha \tag{26}$$

where the confidence limits $\alpha = 99\%$.

### 3.3. Fault Identification Based on the Contribution of $T^2$ Statistic

According to the definition of variable contribution based on CVA proposed by Jiang et al. [36], $T^2$-based variable contribution can be obtained using the following formula:

$$C_{T^2} = T^2 = d^T d = d^T K \hat{y}_{p,k} = \sum_{i=1}^{n} \sum_{j=1}^{q} d_j K_{j,i} \hat{y}_{p,i} = \sum_{i=1}^{n} C_{i,T^2} \tag{27}$$

Here, $C_{i,T^2}$ represents the contribution of variable $\hat{y}_i$ to monitoring statistic $T^2$, and $d_j K_{j,i} \hat{y}_{p,i}$ denotes the contribution of variable $\hat{y}_i$ to the $j$-th typical state variable $z_j$. Ultimately, by utilizing the contribution of each variable $\hat{y}_i$ to $T^2$, and dividing it by the sum of

contributions $C_{T^2}$, the percentage of each contribution can be calculated, thereby identifying the variables related to the fault.

$$P_{i,T^2} = \frac{C_{i,T^2}}{C_{T^2}} \tag{28}$$

## 4. Case Study

In this section, to validate the superiority of the proposed method in the diagnosis of incipient faults in nonlinear dynamic systems, the Gap-MK-DCCA method is used for process monitoring and fault detection on the Tennessee Eastman Process (TEP). Additionally, the experimental results are compared with those of CVA, DCCA, and MK-DCCA to highlight excellent fault diagnosis performance of the Gap-MK-DCCA method.

### *4.1. Process Description*

#### 4.1.1. Model Introduction

TEP, created by the Eastman Chemical Company, serves the purpose of providing a realistic industrial process for the evaluation of process monitoring and fault detection methods. It has been extensively utilized as a benchmark dataset for comparing various methods in the field of process monitoring. The test process is a simulation based on a real industrial process initially proposed by Downs and Vogel [37]. The simulation experiment model and data used in this paper can be found at http://depts.washington.edu/control/LARRY/TE/download.html (accessed on 11 November 2023). The process consists of five main units: reactor, condenser, compressor, separator, and stripper tower. The reactants in the entire chemical process are mainly composed of four gaseous materials, namely $A$, $C$, $D$, and $E$; $G$ and $H$ are two products generated in the reaction, accompanied by a byproduct $F$; in addition, a small amount of inert gas $B$ is present in the product feed. The reaction equation for the whole process is as follows:

$$\begin{aligned} &A(g) + C(g) + D(g) \rightarrow G(liq), \\ &A(g) + C(g) + E(g) \rightarrow H(liq), \\ &A(g) + E(g) \rightarrow F(liq), \\ &3D(g) \rightarrow 2F(liq) \end{aligned} \tag{29}$$

The reaction is irreversible, exothermic, and approximately first-order with respect to reactant concentrations, and the reaction rate is a temperature-dependent function.

#### 4.1.2. Process Variables

The sample data of TEP includes two types of variables: measurement variables represented by XMEAS, a total of 41 variables, as shown in Table 1; and manipulated variables denoted by XMV and a total of 11 variables, as shown in Table 2. In our study, inputs (u) are manipulated variables and outputs (y) are measured variables and system output variables. In addition, all samples were obtained under a 48 h simulation run with 3 min sampling intervals, and faults were introduced after 8 h of model operation, with a total of 960 observations and the first 160 observations being normal data. It is worth mentioning that all process measurements contain Gaussian noise.

**Table 1.** Measurement variables of TEP.

| Variable | Description | Unit |
|---|---|---|
| XMEAS(1) | A feed (stream 1) | $km^3/h$ |
| XMEAS(2) | D feed (stream 2) | kg/h |
| XMEAS(3) | E feed (stream 3) | kg/h |
| XMEAS(4) | Total feed (stream 4) | $km^3/h$ |
| XMEAS(5) | Recirculation flow rate (stream 8) | $km^3/h$ |
| XMEAS(6) | Reactor feed rate (stream 6) | $km^3/h$ |

**Table 1.** *Cont.*

| Variable | Description | Unit |
|---|---|---|
| XMEAS(7) | Reactor pressure | kPa |
| XMEAS(8) | Reactor level | % |
| XMEAS(9) | Reactor temperature | °C |
| XMEAS(10) | Discharge rate (stream 9) | $km^3/h$ |
| XMEAS(11) | Product separator temperature | °C |
| XMEAS(12) | Product separator level | % |
| XMEAS(13) | Product separator pressure | kPa |
| XMEAS(14) | Product separator bottom flow rate (stream 10) | $m^3/h$ |
| XMEAS(15) | Stripper tower level | % |
| XMEAS(16) | Stripper tower level pressure | kPa |
| XMEAS(17) | Stripper tower bottom flow rate (stream 11) | $m^3/h$ |
| XMEAS(18) | Stripper tower temperature | °C |
| XMEAS(19) | Stripper tower flow rate | kg/h |
| XMEAS(20) | Compressor power | kW |
| XMEAS(21) | Reactor cooling water outlet temperature | °C |
| XMEAS(22) | Separator cooling water outlet temperature | °C |
| XMEAS(23) | Component A (stream 6) | mol% |
| XMEAS(24) | Component B (stream 6) | mol% |
| XMEAS(25) | Component C (stream 6) | mol% |
| XMEAS(26) | Component D (stream 6) | mol% |
| XMEAS(27) | Component E (stream 6) | mol% |
| XMEAS(28) | Component F (stream 6) | mol% |
| XMEAS(29) | Component A (stream 9) | mol% |
| XMEAS(30) | Component B (stream 9) | mol% |
| XMEAS(31) | Component C (stream 9) | mol% |
| XMEAS(32) | Component D (stream 9) | mol% |
| XMEAS(33) | Component E (stream 9) | mol% |
| XMEAS(34) | Component F (stream 9) | mol% |
| XMEAS(35) | Component G (stream 9) | mol% |
| XMEAS(36) | Component H (stream 9) | mol% |
| XMEAS(37) | Component D (stream 11) | mol% |
| XMEAS(38) | Component E (stream 11) | mol% |
| XMEAS(39) | Component F (stream 11) | mol% |
| XMEAS(40) | Component G (stream 11) | mol% |
| XMEAS(41) | Component H (stream 11) | mol% |

**Table 2.** Manipulated variables of TEP.

| Variable | Description | Unit |
|---|---|---|
| XMV(1) | D feed flow rate | kg/h |
| XMV(2) | E feed flow rate | kg/h |
| XMV(3) | A feed flow rate | $km^3/h$ |
| XMV(4) | Total Feed flow rate | $km^3/h$ |
| XMV(5) | Compressor recirculation valve | % |
| XMV(6) | Discharge valve | % |
| XMV(7) | Separator tank liquid flow rate | $m^3/h$ |
| XMV(8) | Stripper tower liquid product flow rate | $m^3/h$ |
| XMV(9) | Stripper tower water flow valve | % |
| XMV(10) | Reactor cooling water flow rate | $m^3/h$ |
| XMV(11) | Condenser cooling water flow rate | $m^3/h$ |

#### 4.1.3. Process Faults

The TEP simulation model includes 20 pre-set faults, where faults 1–15 are known faults, and the remaining 5 are unknown faults.

Among the known faults, faults 1–7 are caused by step changes in process variables, faults 9–12 are closely related to the stochastic variation in process variables, fault 13

represents a slow drift in reaction dynamics, while faults 14 and 15 are associated with sticking of the valve. Detailed information about these faults is provided in Table 3.

**Table 3.** Fault information of TEP.

| Fault ID | Fault Description | Fault Type |
|---|---|---|
| 1 | A/C feed ratio, component B is constant (stream 4) | Abrupt |
| 2 | Component B, A/C feed ratio is constant (stream 4) | Abrupt |
| 3 | Feed temperature of D (stream 2) | Abrupt |
| 4 | Reactor cooling water inlet temperature | Abrupt |
| 5 | Condenser cooling water inlet temperature | Abrupt |
| 6 | feed losses of A (stream 1) | Abrupt |
| 7 | Pressure loss in C - reduced availability (Stream 4) | Abrupt |
| 8 | A, B, C feed components (stream 4) | Random variable |
| 9 | Feed temperature of D (stream 2) | Random variable |
| 10 | Feed temperature of C (stream 2) | Random variable |
| 11 | Reactor cooling water inlet temperature | Random variable |
| 12 | Condenser cooling water inlet temperature | Random variable |
| 13 | Reaction dynamics | Incipient |
| 14 | Reactor cooling water valves | Sticking |
| 15 | Condenser cooling water valves | Sticking |
| 16 | unknown | / |
| 17 | unknown | / |
| 18 | unknown | / |
| 19 | unknown | / |
| 20 | unknown | / |

### *4.2. Process Monitoring and Fault Detection*

In this subsection, we evaluate the fault detection performance of the Gap-MK-DCCA method by using samples collected under the normal operation state of the model mentioned in Section 4.1 as the training set, selecting samples collected under fault conditions of the model as the test set, and using manipulated variables and measurement variables as system inputs and outputs, respectively. For two common types of faults, namely abrupt faults and incipient faults, we choose Fault 2 and Fault 13, respectively, for specific case studies. The specific details are shown in Table 4.

**Table 4.** Fault details for detection.

| Fault ID | Fault Variable | Introduction Time (h) | Fault Type | Subspace |
|---|---|---|---|---|
| Fault 2 | XMV(6) | 8th (160th Sampling Point) | Abrupt | Input |
| Fault 13 | / | 8th (160th Sampling Point) | Incipient | Output |

In addition, in order to better evaluate the performance of the proposed method, this study utilizes four metrics—Fault Detection Rate (FDR), False Alarm Rate (FAR), Missed Detection Rate (MDR), and Fault Detection Time (FDT)—for quantification. Fault Detection Time (FDT) represents the sampling moment corresponding to three consecutive statistical indicators exceeding the control limits after a fault occurs. The remaining three indicators are defined as follows:

$$FDR = \frac{TP}{TP + FN} \tag{30}$$

$$FAR = \frac{FP}{FP + TN} \tag{31}$$

$$MDR = \frac{FN}{TP + FN} \tag{32}$$

Here, TP represents the number of faults correctly detected, TN is the number of normals correctly detected, FP denotes the number of normal samples falsely reported as faults, and FN is the number of faults missed and reported as normal.

#### 4.2.1. Abrupt Fault Detection

In the experimental session, considering the commonality of abrupt faults in modern process industries and their relatively lower detection difficulty, we initially select an abrupt fault (Fault 2) for process monitoring and fault detection experiments, and the comparative methods are CVA, DCCA, and MK-DCCA, respectively. These methods employ the traditional z-score data preprocessing method, while Gap-MK-DCCA uses the Gap metric method for data preprocessing. Additionally, all methods use KDE to calculate the corresponding thresholds. The detailed detection results for Fault 2 are presented in Figure 2.

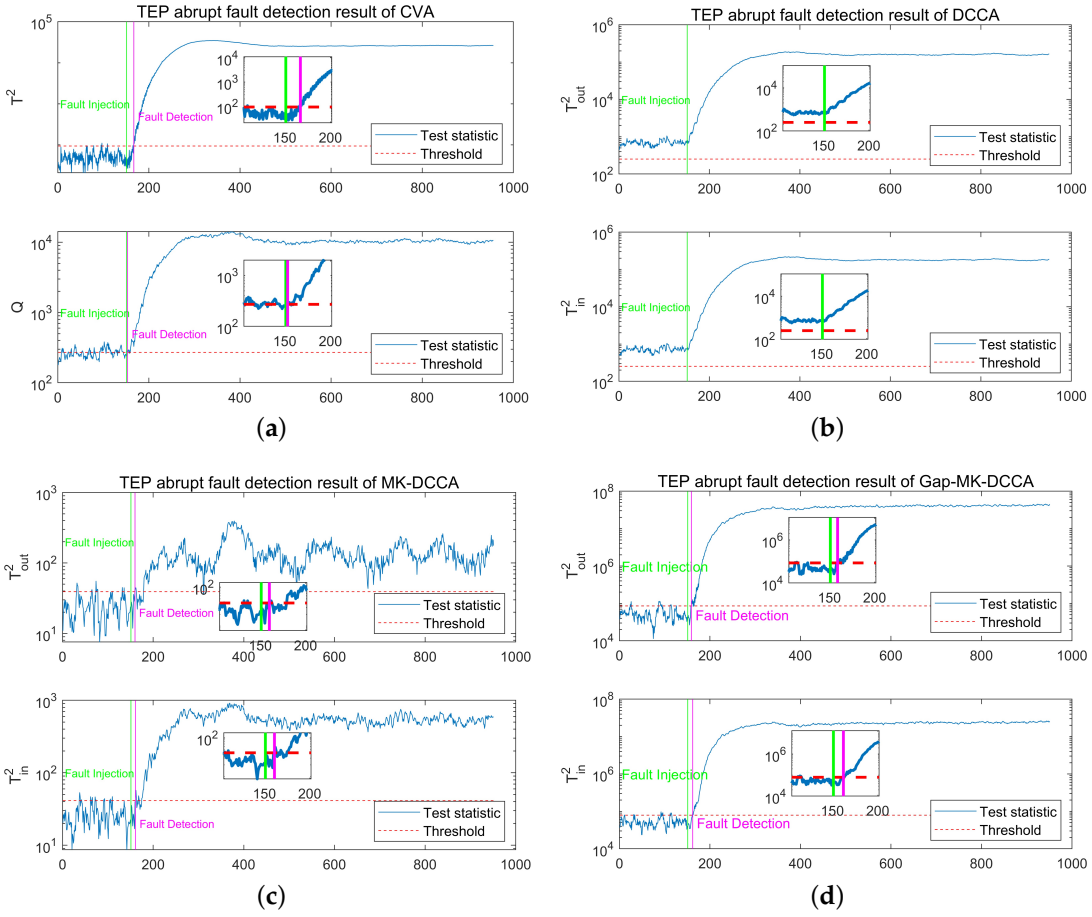


**Figure 2.** Detection results for Fault 2 with different methods: (**a**) Detection results of CVA. (**b**) Detection results of DCCA. (**c**) Detection results of MK-DCCA. (**d**) Detection results of Gap-MK-DCCA.

As can be seen from Figure 2, all four methods can be used to describe the system operation status as well as fault development trends through monitoring indicators for abrupt fault scenarios. Among them, CVA, MK-DCCA, and Gap-MK-DCCA methods can quickly issue warning signals after the fault occurs, while the DCCA method struggles when calculating thresholds for monitoring tasks in nonlinear dynamic systems where the data do not follow a Gaussian distribution, rendering it ineffective for fault detection. This is largely attributed to the inability of the DCCA method to deal with complex system properties, such as non-linearity and non-Gaussianity. Meanwhile, from Figure 2c, it can be observed that the robustness of the MK-DCCA method is poor, being severely affected by process noise and measurement noise. Although the CVA method produces good detection results, it is unable

to explain the dynamic characteristics between system inputs and outputs. However, the Gap-MK-DCCA method achieves a description of dynamic characteristics between system inputs and outputs by separately categorizing process variables into inputs (manipulated variables) and outputs (measured variables). Simultaneously, this method employs the Gap metric for data preprocessing, mitigating various noise interferences by mapping data onto the Riemannian sphere. It also addresses the issue of non-Gaussian data distribution, achieving satisfactory fault detection performance under abrupt fault scenarios.

Simultaneously, to better compare the four methods, the detection performance is quantified using FDT, FAR, MDR, and FDT indicators. Table 5 presents detailed information on the performance indicators for all methods.

**Table 5.** Detection performance indicators for Fault 2 with various methods.

| Fault ID | Method | Statistics | FDR(%) | FAR(%) | MDR(%) | FDT (Sampling Point) |
|---|---|---|---|---|---|---|
| Fault 2 | CVA | $T^2$ | 97.88 | 0 | 2.12 | 167th |
| | | $Q$ | 100 | 22.52 | 0 | 153rd |
| | DCCA | $T^2_{in}$ | 100 | 100 | 0 | 1st |
| | | $T^2_{out}$ | 100 | 100 | 0 | 1st |
| | MK-DCCA | $T^2_{in}$ | 97 | 0.66 | 3 | 161st |
| | | $T^2_{out}$ | 96 | 1.32 | 4 | 160th |
| | Gap-MK-DCCA | $T^2_{in}$ | 98.5 | 0 | 1.5 | 161st |
| | | $T^2_{out}$ | 98.25 | 3.31 | 1.75 | 159th |

Since Fault 2 belongs to faults occurring in the input subspace, as explained in Section 3.2, statistic $T^2_{in}$ has been identified as the best for detecting faults within the input subspace. Therefore, in this analysis, we choose the results of statistic $T^2_{in}$ for comparison. As for the CVA method, due to the high false alarm rate of 22.52% in the detection results of the $Q$ statistic, we choose the results of the $T^2$ statistic for comparison. In addition, due to the ineffective detection of the DCCA method, it is not included in the discussion here.

It is evident that the proposed method achieves the highest fault detection rate of 98.5% in the detection of abrupt faults (fault 2), while the CVA and MK-DCCA methods achieve 97.88% and 97%, respectively. As for the false alarm rate, the proposed method is 0%, lower than the MK-DCCA method (0.66%) and the same as the CVA method (0%). The fault detection time for the Gap-MK-DCCA and MK-DCCA methods are both at the 161st sampling point, while the CVA method is slightly later at the 167th sampling point. Finally, the proposed method maintains the lowest missed detection rate for abrupt faults at 1.5%; the CVA method is next at 2.12%, and MK-DCCA performs the worst in this aspect at 3%.

In summary, in the detection of abrupt faults, the best performance is achieved by the proposed Gap-MK-DCCA method. Therefore, it is reasonable to apply this method to abrupt fault detection in modern process industrial systems.

#### 4.2.2. Incipient Fault Detection

To further evaluate the fault detection performance of the Gap-MK-DCCA method, this study also conducted an experimental analysis on another challenge in the field of process monitoring and fault detection—incipient fault detection. All methods and parameter settings used in this case are identical to those in Section 4.2.1, and the detailed detection results for Fault 13 are shown in Figure 3.

As can be seen from Figure 3, the threshold calculation issue present in the DCCA method under abrupt fault scenarios still exists, and the other three methods can also issue warnings after some time following the occurrence of a fault. In addition, under incipient fault scenarios, all methods can describe the operating status of the system and fault development trends, but the description provided by the Gap-MK-DCCA method is clearer and more intuitive.

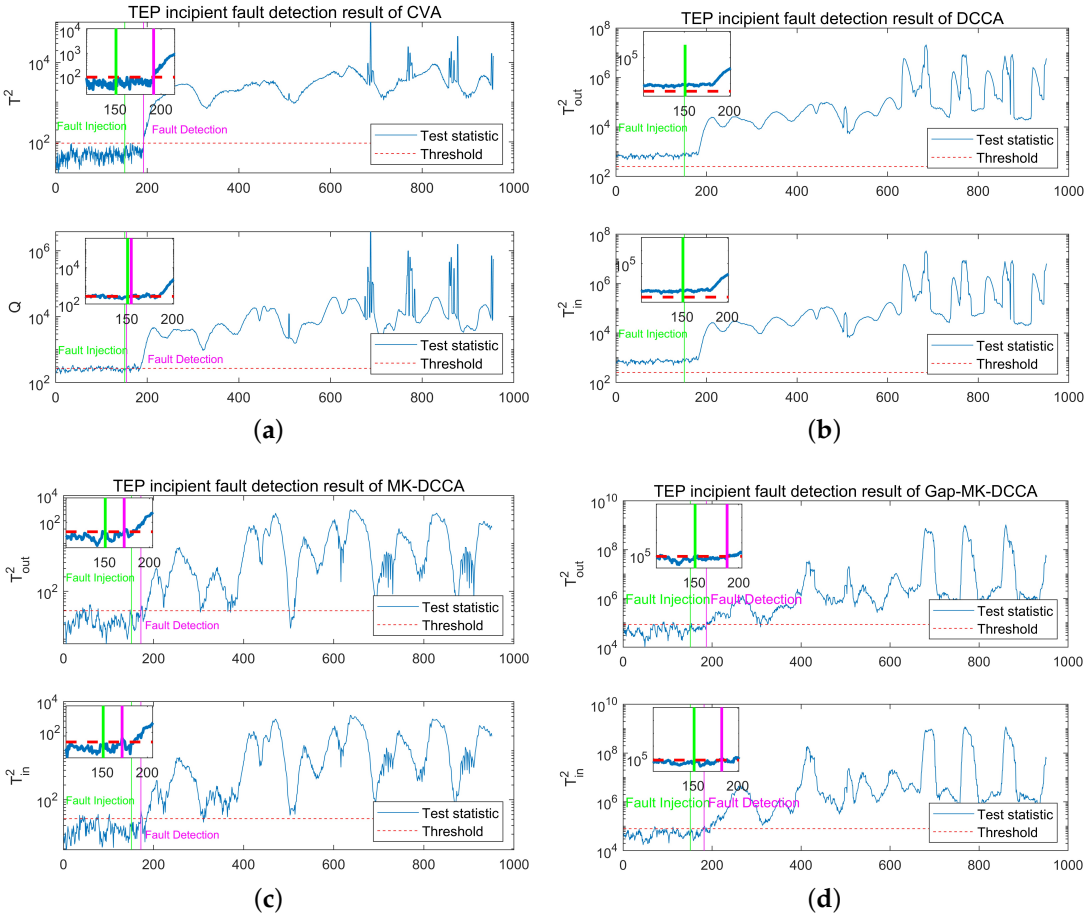


**Figure 3.** Detection results for Fault 13 with different methods: (**a**) detection results of CVA. (**b**) Detection results of DCCA. (**c**) Detection results of MK-DCCA. (**d**) Detection results of Gap-MK-DCCA.

To further analyze and compare the performance of the four detection methods in incipient fault scenarios, FDT, FAR, MDR, and FDT indicators are also used here for quantification. Detailed information on the performance indicators of all methods is shown in Table 6.

**Table 6.** Detection performance indicators for Fault 13 with various methods.

| Fault ID | Method | Statistics | FDR(%) | FAR(%) | MDR(%) | FDT (Sampling Point) |
|---|---|---|---|---|---|---|
| Fault 13 | CVA | $T^2$ | 94 | 0 | 5.25 | 192nd |
| | | $Q$ | 96.5 | 13.91 | 3.5 | 155th |
| | DCCA | $T^2_{in}$ | 100 | 100 | 0 | 1st |
| | | $T^2_{out}$ | 100 | 100 | 0 | 1st |
| | MK-DCCA | $T^2_{in}$ | 95.5 | 0.66 | 4.5 | 172nd |
| | | $T^2_{out}$ | 93 | 1.99 | 7 | 172nd |
| | Gap-MK-DCCA | $T^2_{in}$ | 95 | 2.65 | 5 | 187th |
| | | $T^2_{out}$ | 94.5 | 0.66 | 5.5 | 182nd |

Since fault 13 belongs to a fault that occurs in the output subspace, the results of statistic $T^2_{out}$ are selected for analysis and comparison. As for the CVA method, the detection result based on the $Q$ statistic has a high false alarm rate of 13.91%, so the results based on statistic $T^2$ are chosen for comparison. In addition, due to the ineffective detection of the DCCA method, it is not included in the discussion.

From Table 6, it can be seen that in incipient faults scenarios, the MK-DCCA method takes the shortest time to detect faults at the 172nd sampling point, followed by the Gap-MK-DCCA method, which issues a warning signal at the 182nd sampling point, and the CVA method takes the longest at the 192nd sampling point. Thanks to the longest fault detection time, the CVA method achieved zero false alarms, and the false alarm rate of the Gap-MK-DCCA method is 0.66%, much lower than 1.99% of the MK-DCCA method. In addition, in terms of missed detection rate, the CVA method maintains the lowest rate, followed by the Gap-MK-DCCA method, and the MK-DCCA method has the highest missed detection rate at 7%; Finally, in terms of fault detection rate, the Gap-MK-DCCA method still achieves a satisfactory performance, at 94.5%, slightly higher than 94% of the CVA method and 93% of the MK-DCCA method.

In a comprehensive comparison, the performance of the Gap-MK-DCCA method proposed in this paper is still more satisfactory. In conclusion, this method can provide a new perspective for process monitoring and fault detection in incipient fault scenarios.

#### 4.2.3. Other Faults Detection

In addition, in order to further enhance the persuasiveness of the proposed method in terms of fault detection performance, we also conducted fault detection experiments with one randomly selected fault from each of the two types of random variable faults and unknown faults. The specific results of the detection are shown in Figures 4 and 5.

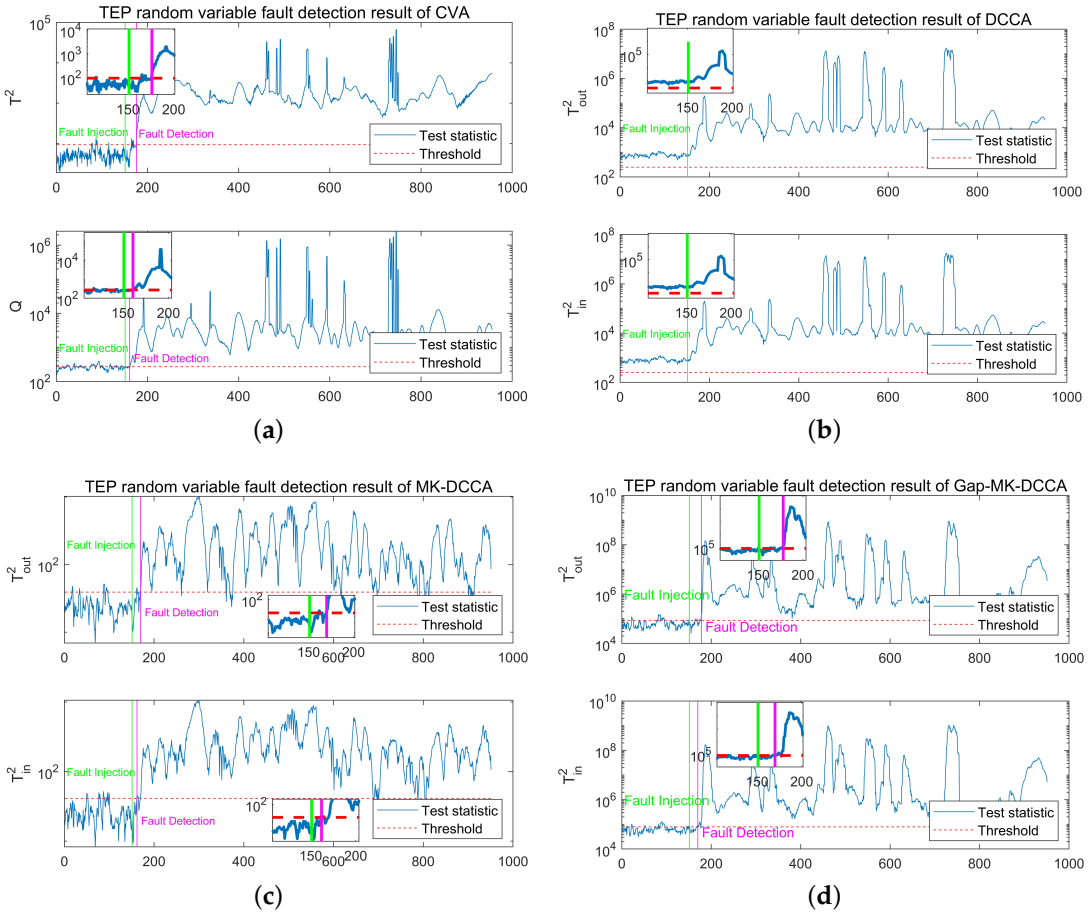


**Figure 4.** Detection results for Fault 8 with different methods: (**a**) detection results of CVA. (**b**) Detection results of DCCA. (**c**) Detection results of MK-DCCA. (**d**) Detection results of Gap-MK-DCCA.

The specific performance metrics associated with these two faults are shown in Tables 7 and 8.

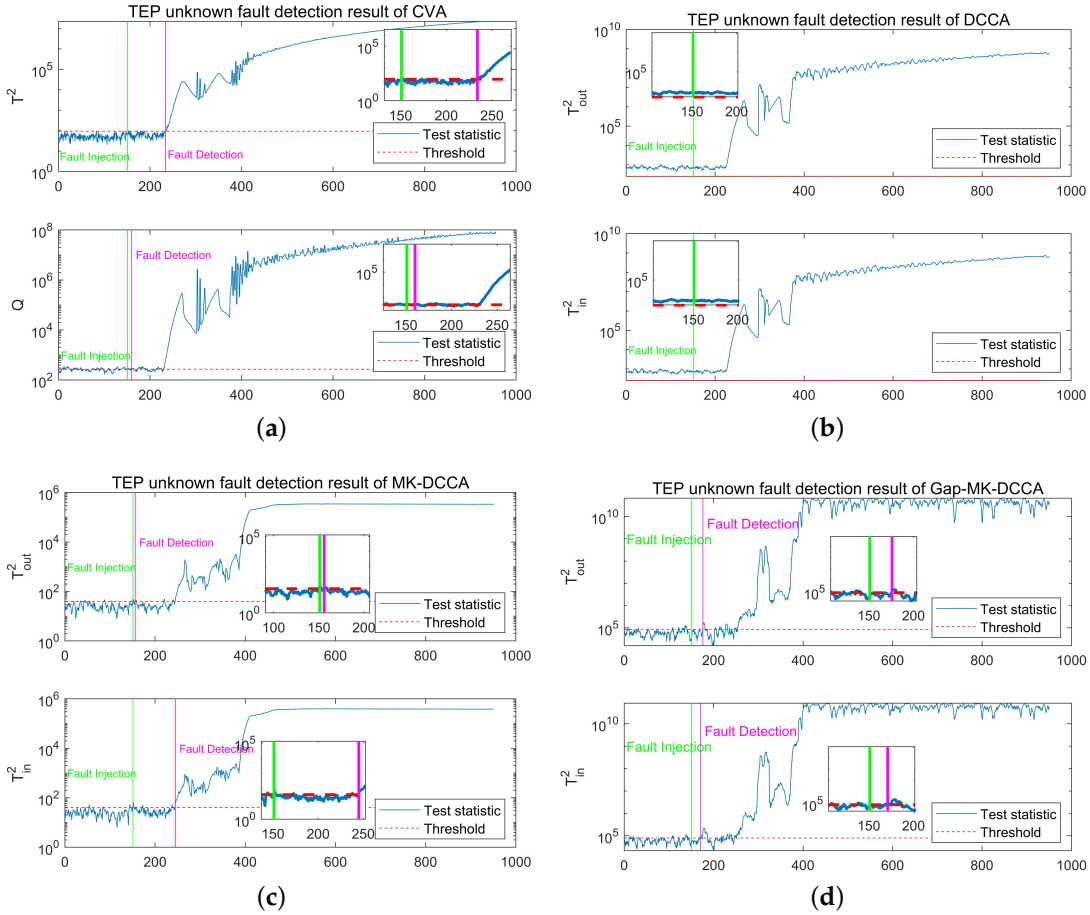


**Figure 5.** Detection results for Fault 18 with different methods: (**a**) detection results of CVA. (**b**) Detection results of DCCA. (**c**) Detection results of MK-DCCA. (**d**) Detection results of Gap-MK-DCCA.

**Table 7.** Detection performance indicators for Fault 8 with various methods.

| Fault ID | Method | Statistics | FDR (%) | FAR (%) | MDR (%) | FDT (Sampling Point) |
|---|---|---|---|---|---|---|
| Fault 8 | CVA | $T^2$ | 96.75 | 1.32 | 3.25 | 176th |
| | | $Q$ | 98.5 | 17.88 | 1.5 | 161st |
| | DCCA | $T^2_{in}$ | 100 | 100 | 0 | 1st |
| | | $T^2_{out}$ | 100 | 100 | 0 | 1st |
| | MK-DCCA | $T^2_{in}$ | 95.63 | 1.32 | 4.37 | 162nd |
| | | $T^2_{out}$ | 91 | 3.31 | 9 | 170th |
| | Gap-MK-DCCA | $T^2_{in}$ | 97.13 | 5.96 | 2.87 | 170th |
| | | $T^2_{out}$ | 96.5 | 7.28 | 3.5 | 178th |

**Table 8.** Detection performance indicators for Fault 18 with various methods.

| Fault ID | Method | Statistics | FDR (%) | FAR (%) | MDR (%) | FDT (Sampling Point) |
|---|---|---|---|---|---|---|
| Fault 18 | CVA | $T^2$ | 89.5 | 0 | 10.5 | 234th |
| | | $Q$ | 92.25 | 19.21 | 7.75 | 160th |
| | DCCA | $T^2_{in}$ | 100 | 100 | 0 | 1st |
| | | $T^2_{out}$ | 100 | 100 | 0 | 1st |
| | MK-DCCA | $T^2_{in}$ | 88.13 | 0 | 11.87 | 245th |
| | | $T^2_{out}$ | 88.75 | 0.66 | 11.25 | 156th |
| | Gap-MK-DCCA | $T^2_{in}$ | 89.88 | 2.65 | 10.12 | 171st |
| | | $T^2_{out}$ | 88.13 | 2.65 | 11.87 | 176th |

### *4.3. Fault Identification*

In the previous section, the performance of the proposed method in fault detection has been verified. In this section, we mainly analyze the accuracy of the fault identification method based on the contribution of the $T^2$ statistic. Since the variable related to Fault 13 is not clear, here we initially identify Fault 2, and the contribution of relevant variables is shown in Figure 6.

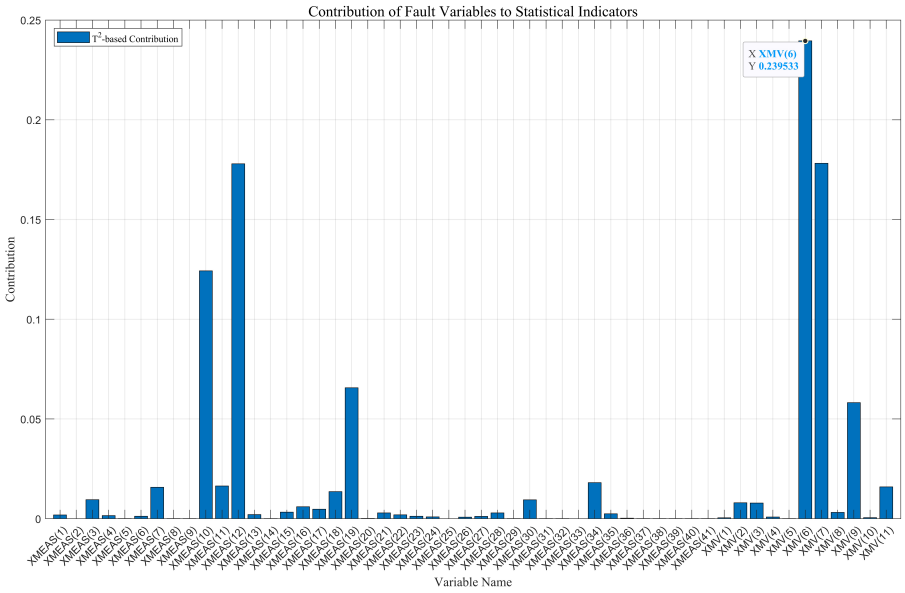


**Figure 6.** Contribution of relevant variables for Fault 2.

It is evident that variable XVM(6) has the highest contribution to fault statistics, accounting for 23.95%, which is consistent with the corresponding fault information in Table 4, successfully verifying the effectiveness of the fault identification method used in this research. Subsequently, we further carried out fault identification for Fault 13, and the specific results are shown in Figure 7.

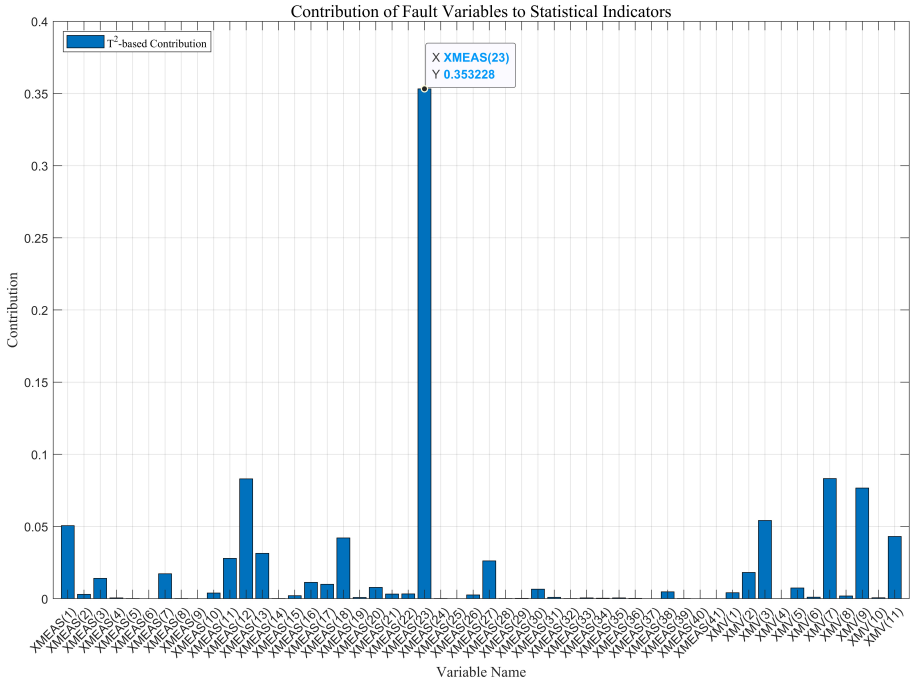


**Figure 7.** Contribution of relevant variables for Fault 13.

It is clear that for Fault 13, variable XMEAS(23) contributes the most to fault statistics, suggesting that it may be the process variable causing the fault, which can be localized to the fault site based on the location of the corresponding sensor.

## 5. Conclusions

This paper emphasizes the importance of fault detection in modern process industrial systems and proposes an intelligent process monitoring and fault detection method based on Gap-MK-DCCA. By employing the Gap metric for data preprocessing, it overcomes the problem of unsatisfactory detection performance of traditional methods when the process data do not follow the Gaussian distribution. This approach inherits the capability of the MK-DCCA method in handling complex characteristics of the system, such as nonlinearity and dynamics in modern process industry fault diagnosis. Simultaneously, it introduces the KDE method for threshold computation, addressing the challenge that statistics do not follow the Gaussian distribution, reducing model parameters, and improving the adaptability of thresholds. Finally, a case study is conducted on the industry-recognized Tennessee Eastman Process benchmark dataset. Comparative experiments with CVA, DCCA, and MK-DCCA methods validate the proposed approach for effective fault detection in process monitoring of modern process industrial systems, particularly in scenarios involving abrupt and incipient faults.

Nevertheless, there are some limitations in this study. Although the use of the KDE method for threshold calculation partially reduces the parameters of the fault detection model, the detection performance still significantly depends on the selection of the kernel function and kernel parameters in the kernel method. Furthermore, the choice of the lead parameter $f$, lag parameter $p$, and the number of singular values $n$ also plays a vital role in influencing detection performance. Although grid search can identify optimal parameters, it undoubtedly increases workload and computational complexity, leading to poor generalization of the method to different detection objects. Consequently, how to adaptively choose optimal parameters to achieve a lightweight detection model and enhance generalization performance remains a pressing issue for future work.

**Author Contributions:** Conceptualization, J.W. and M.Z.; methodology, J.W. and M.Z.; validation, J.W. and L.C.; formal analysis, J.W.; investigation, J.W. and L.C.; resources, M.Z.; data curation, L.C.; writing—original draft preparation, J.W.; writing—review and editing, M.Z. and J.W.; visualization, J.W.; supervision, M.Z.; project administration, M.Z.; funding acquisition, M.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China, (Grant No. 62003106) and the Provincial Natural Science Foundation of Guizhou Province, China (Grant No. ZK (2021) 321. and [2017]5788).

**Data Availability Statement:** The data that support the findings of this study are available from the corresponding author upon reasonable request.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Qin, S.J. Survey on data-driven industrial process monitoring and diagnosis. *Annu. Rev. Control* **2012**, *36*, 220–234. [CrossRef]
2. De Carvalho Michalski, M.A.; de Souza, G.F.M. Comparing PCA-based fault detection methods for dynamic processes with correlated and Non-Gaussian variables. *Expert Syst. Appl.* **2022**, *207*, 117989. [CrossRef]
3. Nawaz, M.; Maulud, A.S.; Zabiri, H.; Taqvi, S.A.A.; Idris, A. Improved process monitoring using the CUSUM and EWMA-based multiscale PCA fault detection framework. *Chin. J. Chem. Eng.* **2021**, *29*, 253–265. [CrossRef]
4. Zhang, S.; Wang, S. Spectral radius-based interval principal component analysis (SR-IPCA) for fault detection in industrial processes with imprecise data. *J. Process Control* **2022**, *114*, 105–119. [CrossRef]
5. Zhou, W.; Hou, J. Implementation of fault isolation for molten salt reactor using PCA and contribution analysis. *Ann. Nucl. Energy* **2022**, *173*, 109138. [CrossRef]
6. Bakdi, A.; Kouadri, A.; Bensmail, A. Fault detection and diagnosis in a cement rotary kiln using PCA with EWMA-based adaptive threshold monitoring scheme. *Control Eng. Pract.* **2017**, *66*, 64–75. [CrossRef]

7. Lu, C.; Zeng, J.; Dong, Y.; Xu, X. Streaming variational probabilistic principal component analysis for monitoring of nonstationary process. *J. Process Control* **2024**, *133*, 103134. [CrossRef]
8. Ding, S.X.; Yin, S.; Peng, K.; Hao, H.; Shen, B. A novel scheme for key performance indicator prediction and diagnosis with application to an industrial hot strip mill. *IEEE Trans. Ind. Inform.* **2012**, *9*, 2239–2247. [CrossRef]
9. Lin, X.; Sun, R.; Wang, Y. Improved key performance indicator-partial least squares method for nonlinear process fault detection based on just-in-time learning. *J. Frankl. Inst.* **2023**, *360*, 1–17. [CrossRef]
10. Zhou, P.; Zhang, R.; Liang, M.; Fu, J.; Wang, H.; Chai, T. Fault identification for quality monitoring of molten iron in blast furnace ironmaking based on KPLS with improved contribution rate. *Control Eng. Pract.* **2020**, *97*, 104354. [CrossRef]
11. Qin, Y.; Lou, Z.; Wang, Y.; Lu, S.; Sun, P. An analytical partial least squares method for process monitoring. *Control Eng. Pract.* **2022**, *124*, 105182. [CrossRef]
12. Muradore, R.; Fiorini, P. A PLS-based statistical approach for fault detection and isolation of robotic manipulators. *IEEE Trans. Ind. Electron.* **2011**, *59*, 3167–3175. [CrossRef]
13. Kong, X.; Luo, J.; Feng, X.; Liu, M. A General Quality-Related Nonlinear Process Monitoring Approach Based on Input–Output Kernel PLS. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 1–12. [CrossRef]
14. Ruiz-Cárcel, C.; Cao, Y.; Mba, D.; Lao, L.; Samuel, R. Statistical process monitoring of a multiphase flow facility. *Control Eng. Pract.* **2015**, *42*, 74–88. [CrossRef]
15. Pilario, K.E.S.; Cao, Y. Canonical variate dissimilarity analysis for process incipient fault detection. *IEEE Trans. Ind. Inform.* **2018**, *14*, 5308–5315. [CrossRef]
16. Pilario, K.E.S.; Cao, Y.; Shafiee, M. Incipient fault detection, diagnosis, and prognosis using canonical variate dissimilarity analysis. In *Computer Aided Chemical Engineering*; Elsevier: Amsterdam, The Netherlands, 2019; Volume 46, pp. 1195–1200.
17. Pilario, K.E.S.; Cao, Y.; Shafiee, M. Mixed kernel canonical variate dissimilarity analysis for incipient fault monitoring in nonlinear dynamic processes. *Comput. Chem. Eng.* **2019**, *123*, 143–154. [CrossRef]
18. Chen, Z.; Ding, S.X.; Zhang, K.; Li, Z.; Hu, Z. Canonical correlation analysis-based fault detection methods with application to alumina evaporation process. *Control Eng. Pract.* **2016**, *46*, 51–58. [CrossRef]
19. Chen, Z.; Liang, K. Canonical correlation analysis–based fault diagnosis method for dynamic processes. In *Fault Diagnosis and Prognosis Techniques for Complex Engineering Systems*; Elsevier: Amsterdam, The Netherlands, 2021; pp. 51–88.
20. Chen, Z.; Deng, Q.; Zhao, Z.; Tang, P.; Luo, W.; Liu, Q. Application of just-in-time-learning CCA to the health monitoring of a real cold source system. *IFAC-PapersOnLine* **2022**, *55*, 23–30. [CrossRef]
21. Gao, L.; Li, D.; Yao, L.; Gao, Y. Sensor drift fault diagnosis for chiller system using deep recurrent canonical correlation analysis and k-nearest neighbor classifier. *ISA Trans.* **2022**, *122*, 232–246. [CrossRef]
22. Li, X.; Wang, Y.; Tang, B.; Qin, Y.; Zhang, G. Canonical correlation analysis of dimension reduced degradation feature space for machinery condition monitoring. *Mech. Syst. Signal Process.* **2023**, *182*, 109603. [CrossRef]
23. Peng, X.; Ding, S.X.; Du, W.; Zhong, W.; Qian, F. Distributed process monitoring based on canonical correlation analysis with partly-connected topology. *Control Eng. Pract.* **2020**, *101*, 104500. [CrossRef]
24. Wang, S.; Zhou, C.; Cao, Y.; Yang, S.H. Multiblock dynamic enhanced canonical correlation analysis for industrial MSW combustion state monitoring. *Control Eng. Pract.* **2023**, *138*, 105612. [CrossRef]
25. Luo, L.; Wang, W.; Bao, S.; Peng, X.; Peng, Y. Robust and sparse canonical correlation analysis for fault detection and diagnosis using training data with outliers. *Expert Syst. Appl.* **2024**, *236*, 121434. [CrossRef]
26. Wang, S.; Ju, Y.; Fu, C.; Xie, P.; Cheng, C. A Reversible Residual Network-Aided Canonical Correlation Analysis to Fault Detection and Diagnosis in Electrical Drive Systems. *IEEE Trans. Instrum. Meas.* **2024**, 1. [CrossRef]
27. Gao, L.; Li, D.; Chen, Z.; Ding, S.X.; Luo, H. SIR-Aided Dynamic Canonical Correlation Analysis for Fault Detection and Isolation of Industrial Automation Systems. *IEEE Trans. Ind. Electron.* **2023**, 1–11. [CrossRef]
28. Xiu, X.; Li, Y. Learning Sparse Kernel CCA With Graph Priors for Nonlinear Process Monitoring. *IEEE Sens. J.* **2023**, *23*, 7381–7389. [CrossRef]
29. Wu, J.; Zhang, M.; Chen, L. MK-DCCA-Based Fault Diagnosis for Incipient Faults in Nonlinear Dynamic Processes. *Processes* **2023**, *11*, 2927. [CrossRef]
30. Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*; Cambridge University Press: Cambridge, UK, 2000.
31. Smola, A.; Ovári, Z.; Williamson, R.C. Regularization with dot-product kernels. *Adv. Neural Inf. Process. Syst.* **2000**, *13*, 290–296.
32. Jordaan, E. Development of Robust Inferential Sensors: Industrial Applications of Support Vector Machines for Regression. Ph.D Thesis, Technische Universiteit Eindhoven, Eindhoven, The Netherlands, 2002. [CrossRef]
33. Negiz, A.; Çlinar, A. Statistical monitoring of multivariable dynamic processes with state-space models. *AIChE J.* **1997**, *43*, 2002–2020. [CrossRef]
34. Martin, E.; Morris, A. Non-parametric confidence bounds for process performance monitoring charts. *J. Process Control* **1996**, *6*, 349–358. [CrossRef]
35. Chen, Q.; Goulding, P.; Sandoz, D.; Wynne, R. The application of kernel density estimates to condition monitoring for process industries. In Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No.98CH36207), Philadelphia, PA, USA, 26–26 June 1998; Volume 6, pp. 3312–3316.

36. Jiang, B.; Huang, D.; Zhu, X.; Yang, F.; Braatz, R.D. Canonical variate analysis-based contributions for fault identification. *J. Process Control* **2015**, *26*, 17–25. [CrossRef]
37. Downs, J.J.; Vogel, E.F. A plant-wide industrial process control problem. *Comput. Chem. Eng.* **1993**, *17*, 245–255. [CrossRef]