





## Article

# Fault Detection Method via $k$ -Nearest Neighbor Normalization and Weight Local Outlier Factor for Circulating Fluidized Bed Boiler with Multimode Process

Minseok Kim <sup>1</sup>, Seunghwan Jung <sup>1</sup>, Baekcheon Kim <sup>1</sup>, Jinyong Kim <sup>1</sup>, Eunkyeong Kim <sup>1</sup>, Jonggeun Kim <sup>2</sup>  
and Sungshin Kim <sup>1,\*</sup>

<sup>1</sup> Department of Electrical and Electronics Engineering, Pusan National University, Busan 46241, Korea

<sup>2</sup> Artificial Intelligence Research Center, Korea Electrotechnology Research Institute, Changwon 51543, Korea

\* Correspondence: sskim@pusan.ac.kr; Tel.: +82-515102374

**Abstract:** In modern complex industrial processes, mode changes cause unplanned shutdowns, potentially shortening the lifespan of key equipment and incurring significant maintenance costs. To avoid this problem, a method that can detect the fault of equipment operating in various modes is required. Therefore, we propose a novel fault detection method that uses the  $k$ -nearest neighbor normalization-based weight local outlier factor (WLOF). The proposed method performs local normalization using neighbors to consider possible mode changes in the normal data and WLOF is used for fault detection. In contrast to statistical methods, such as principal component analysis (PCA) and independent component analysis (ICA), the local outlier factor (LOF) uses the density of neighbors. However, because LOF is significantly affected by the distance between its neighbors, the weight is multiplied proportionally to the distance between each neighbor to improve the fault detection performance of the LOF. The efficiency of the proposed method was evaluated using a multimode numerical case and a circulating fluidized bed boiler. The experimental results show that the proposed method outperforms conventional PCA, kernel PCA (KPCA),  $k$ -nearest neighbor ( $k$ NN), and LOF. In particular, the proposed method improved the detection accuracy by 20% compared with conventional methods. Therefore, the proposed method can be applied to a real process operating in multiple modes.

**Keywords:** fluidized bed boiler; fault detection; weighted normalization; local outlier factor



**Citation:** Kim, M.; Jung, S.; Kim, B.; Kim, J.; Kim, E.; Kim, J.; Kim, S. Fault Detection Method via  $k$ -Nearest Neighbor Normalization and Weight Local Outlier Factor for Circulating Fluidized Bed Boiler with Multimode Process. *Energies* **2022**, *15*, 6146.

<https://doi.org/10.3390/en15176146>

en15176146

Academic Editors: Jose A. Almendros-Ibanez and Daniel Morinigo-Sotelo

Received: 18 July 2022

Accepted: 20 August 2022

Published: 24 August 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Modern complex industrial processes (e.g., power plants, chemical and manufacturing processes) include different modes of operation owing to different market requirements, product specification changes, manufacturing strategy changes, or other reasons [1–3]. For example, if the production specifications of a process are changed, various operating conditions, such as the reactor temperature, pressure, feed flow rate, catalyst compositions, and feedstock quality, must be adjusted to fit the operating mode [4]. Changing the operating strategy according to the supply and demand of products can improve customer satisfaction, product quality, and process efficiency. However, fault detection and diagnosis of a multimode process are challenging because the process becomes more complex and automatic due to changes in the conditions suitable for multiple operations, such as the temperature, pressure, and feed flow [5]. Faults are caused by various factors, such as facility defects, construction defects, and malfunctions. If the fault is neglected without an appropriate act, it will not only shorten the life of key facilities (e.g., turbines, compressors, generators, etc.) but also cause enormous economic losses such as repair and replacement costs. Consequently, an initial fault can affect other normal equipment, eventually leading to a fault. To overcome the problem, fault detection and diagnosis (FDD), which identifies the status of the current system, is essential. Because early fault

detection can prevent unplanned shutdowns and huge maintenance cost problems, many scholars have recently proposed various FDD methods by analyzing multimode processes from a different perspective [6–8].

The detection of faults in various industrial processes has been studied, including PCA-based photovoltaic system monitoring [9] and support vector regression-based robot swarm system monitoring [10]. ANN-based tube leak detection [11], water-cooling wall tube leak detection using the three-dimensional space-location method [12], phase transformation estimator-based tube leak detection [13], and ANN-based air heat exchanger modeling [14] have been proposed in various studies. As in the above cases, many studies have been conducted to detect a fault (e.g., tube leak, air heat exchanger, etc.) in industrial process; however, few studies have been conducted on the multimode of fluidized bed boilers. In addition, model-based methods require various types of faulty data to obtain process-related knowledge and fault information [15]. Compared with these methods, the proposed method uses only normal data for fault detection and relatively little training data is required. In addition, data labeling according to each mode is not required because it is transformed into a single mode using local normalization and then detects a fault.

Multimode fault detection can be classified into single models and multiple models [16]. The single model can explain other specific modes using one model. Ma et al. [5] proposed a local neighborhood standardization strategy that allowed multimodal data to follow a single distribution. Ge et al. [17] developed a local model approach that builds a dynamic model based on queries using data stored in a database. Guo et al. [18] proposed a novel fault detection method based on weighted difference principal component analysis (WDPCA) for monitoring multimode processes. Yu [2,8] proposed a method for monitoring non-Gaussian processes using a Gaussian mixture model (GMM). Rashid and Yu [19] used a hidden Markov model (HMM) to estimate the distribution of normal operation data with multimodal characteristics. GMM and HMM assume that each mode of data follows a multivariate Gaussian distribution, although the industrial data distribution contains both Gaussian and non-Gaussian characteristics simultaneously [2]. Zhao et al. [20] proposed a multiple PCA method using the minimum squared prediction error (SPE) to select a specific local model. Zhu et al. [21] used the expectation-maximization (EM) method to handle multimodal features. However, the model parameters obtained by EM can often be trapped in a local minimum depending on the initial starting point, and the number of clusters must be predefined before determining the parameters.

The multiple model method selects an appropriate model that can identify different modes or construct multiple models corresponding to different modes. Scott et al. [22,23] used aggregated  $k$ -means clustering to identify the multiple modes. Cai et al. [24] proposed an integrated  $k$ -means clustering method to improve the clustering efficiency in high-dimensional space. Ge et al. [25] used external analysis (EA) to remove unnecessary external variables for multimode monitoring. Zhu et al. [26] proposed a method to compare the similarity between modes using  $k$ -nearest neighbor-based independent component analysis and the principal component analysis ( $k$ -ICA-PCA) clustering algorithm. Song et al. [16] identified a mode using a moving window and LOF. Feng Jian et al. [27] proposed a multirate sampling method that divides data into groups of the same length to detect failures in the data with different sampling lengths. In addition, many studies have been conducted on adaptive PCA, such as overlapping PCA, sub-PCA, super-PCA, and multiple PCA for multimode monitoring [28–32]. Nevertheless, PCA-based methods have performance limitations because they are effective only when the data of each mode are close to one another and show weak nonlinearities [8,33].

In general, the characteristics of industrial process data such as the mean and covariance change significantly when one mode changes to another [2,20]. Although different modes can be identified based on this statistical information, multiple model methods have the following disadvantages: first, the mean and covariance structures corresponding to all modes cannot be captured [5]; second, it is difficult to obtain process-related prior knowledge that can be divided into different modes; third, the model should be

updated according to the process mode change; and fourth, the accuracy of the multimodal identification model has a significant impact on the number of clusters.

In this study, we proposed a novel  $k$ -nearest neighbor normalization-based weighted local outlier factor ( $k$ NS-WLOF) to detect faults in a multimode process. The proposed method uses the Euclidean distance to determine the degree of each neighbor to prevent samples close to the normal data from affecting the LOF calculation. The sum of the weights of each neighbor is 1, and the smaller weights are assigned to samples closer to the query vector. Neighbors greater than the average distance of all neighbors are re-adjusted by multiplying the distance by a weight. Therefore, the detection performance of conventional LOF can be improved by assigning weights according to the distance of neighbors. As shown in Figure 1, the procedure of the proposed method is divided into off-line monitoring, which sets a threshold value using training data for fault detection, and on-line monitoring, which detects the fault of a query vector measured in real time. First, local normalization was performed using  $k$ NN to remove the multimode characteristics of the training data.  $k$ NN-based normalization can solve the problem in which the mean and covariance of data differ according to the mode [1,2,34]. Local normalization was performed using  $k$ NN, and the fault in the multimode process was detected using a WLOF. LOF is a quantitative measure of the distance between the fault data and the surrounding neighborhood [35]. This method is intuitive compared to probabilistic methods and is applicable to nonlinear systems and time-varying processes because it does not assume a specific distribution of data such as PCA or ICA. Lee et al. [36] validated that a fault can be detected regardless of the distribution of the LOF, such as a Gaussian mixture distribution or a gamma distribution. Although supervised learning methods, such as the decision tree, random forest, and deep neural network (DNN), can provide excellent results for fault detection, faulty data are generally difficult to obtain in real-world applications and costly. LOF, which belongs to the  $k$ NN-based method, can detect a fault using only normal data; it is more applicable than supervised learning methods. However, the conventional LOF is affected by the distance between neighbors owing to the characteristic of using the density. For example, although it is fault data, it may be considered normal by a neighbor that is particularly close. Therefore, we proposed a method for improving conventional LOF by re-adjusting the distance of each neighbor by assigning a weight to each neighbor.

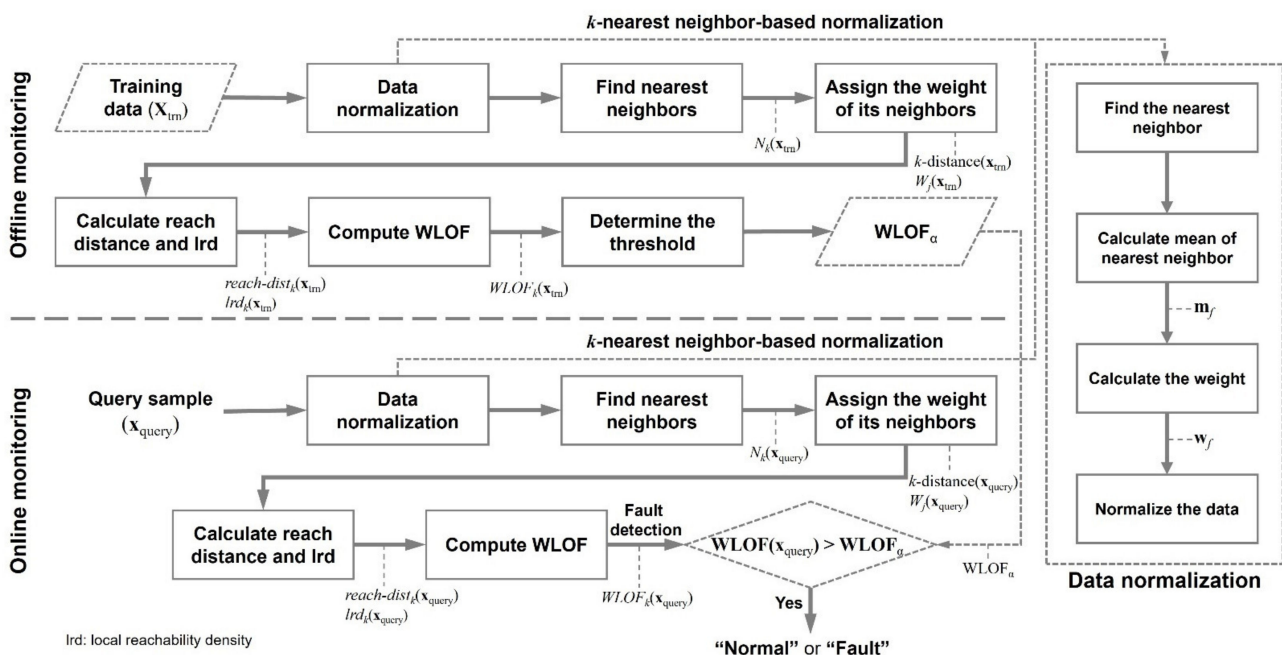


Figure 1. Fault detection procedure using the proposed method.

The detailed procedure of  $k$ NN-WLOF is described in Section 2, and the highlighted advantages of  $k$ NN-WLOF are as follows. First, it does not require prior knowledge of the multimode process. Second, it is not necessary to build multiple regional models based on multimodal characteristics. Third, it is not necessary to assume a specific data distribution (Gaussian or non-Gaussian). It is possible to effectively detect the normal and fault data with adjacent neighbors by assigning a weight according to the distance between them. Finally, compared to DNN, which are recently applied for fault detection, a fault can be detected without acquiring faulty data. To verify the performance, the proposed and comparison methods are applied to two types of multimode cases and a circulating fluidized bed boiler.

The remainder of this paper is organized as follows. Section 2 explains the fault detection using  $k$ NS-WLOF, and Section 3 describes the experimental data used to verify the performance and threshold setting for fault detection. Then, presents the experimental results and discussion. Finally, Section 4 presents the conclusions and future work.

### 2. $k$ NS-WLOF-Based Fault Detection

In this section, the fault detection method using the proposed method is introduced in detail, and the concept of the LOF and the implementation procedure of the WLOF are explained. Subsequently, local normalization using the  $k$ NS strategy and threshold setting for fault detection are described.

#### 2.1. Weighted Local Outlier Factor

LOF is a method for determining outliers beyond a certain distance from the perspective of normal data. Because this method determines whether outliers are based on the density of neighbors, it can also be applied to processes such as nonlinear systems, multimode, and time-varying processes. In particular, LOF is widely applied to outlier detection fields, such as fraud detection and intrusion detection, because it can detect local and global outliers [37]. The detailed concepts of LOF can be found in Breunig et al. [35]. In this study, we proposed a WLOF that assigns weights to each neighbor to improve the conventional LOF. As shown in Figure 2, the calculation procedure is performed as follows: (1) neighbor search using the Euclidean distance; (2)  $k$ -distance and weight assignment of selected neighbors; (3) the reachability distance of the query vector and its neighbors; and (4) calculation of the local reachability density and LOF.

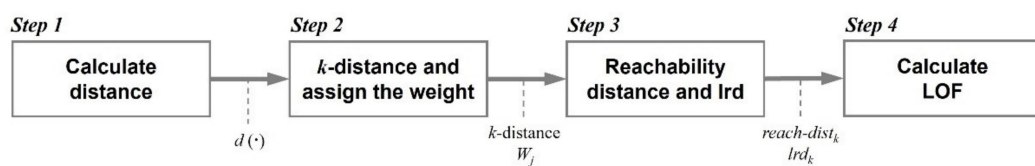


Figure 2. Procedure for calculating LOF.

1. We assumed that the training data ( $\mathbf{X}_{\text{trn}} \in \mathbb{R}^{n \times m}$  with  $n$  samples and  $m$  variables) were collected from a target system operating under normal conditions. The query sample ( $\mathbf{x}_{\text{query}} \in \mathbb{R}^{1 \times m}$  with  $m$  variables) calculates the similarity with  $\mathbf{X}_{\text{trn}}$  using the Euclidean distance as in Equation (1), and then selects  $k$  neighbors of  $\mathbf{X}_{\text{trn}}$  close to  $\mathbf{x}_{\text{query}}$ :

$$d(\mathbf{x}_{\text{trn}}, \mathbf{x}_{\text{query}}) = \sqrt{(x_{\text{trn}}^1 - x_{\text{query}}^1)^2 + (x_{\text{trn}}^2 - x_{\text{query}}^2)^2 + \dots + (x_{\text{trn}}^m - x_{\text{query}}^m)^2} \quad (1)$$

2. The  $k$ -distance means the radius of its local neighborhood ( $N_k(\mathbf{x}_{\text{query}})$ ) of  $\mathbf{x}_{\text{query}}$ , and it is the  $k$ th distance value after sorting the distance values calculated in Equation (1) in ascending order. Given a  $k$ -distance, the neighbors of  $\mathbf{x}_{\text{query}}$  satisfy Equation (2):

$$N_k(\mathbf{x}_{\text{query}}) = \left\{ \mathbf{x}_{\text{trn}} \in \mathbf{X}_{\text{trn}} \setminus \{\mathbf{x}_{\text{trn}}\} \mid d(\mathbf{x}_{\text{trn}}, \mathbf{x}_{\text{query}}) \leq d_k(\mathbf{x}_{\text{trn}}, \mathbf{x}_{\text{query}}) \right\}, \quad (2)$$

where the  $k$ -distance is the  $k$ th neighbor of the query sample. LOF is significantly affected by the distance because it calculates the density using neighbors. To address this limitation, as in Equation (3), a weight ( $W_j(\mathbf{x}_{\text{query}})$ ) is assigned to each neighbor to readjust the distance between the neighbors:

$$W_j(\mathbf{x}_{\text{query}}) = \frac{1}{d_j} / \sum_{i=1}^k \frac{1}{d_i}, \text{ where } j = 1, 2, \dots, k, \tag{3}$$

where  $W_j(\mathbf{x}_{\text{query}})$  is calculated as a value between zero and one, and the sum of the weights is one. If there are more neighbors greater than the average distance of the query sample, the penalty is calculated by multiplying only the weight greater than the average distance calculated in Equation (3). Consequently, neighbors smaller than the average distance are reduced to a distance nearer than the current distance by dividing the weight calculated in Equation (3). By contrast, neighbors farther than the average distance are readjusted to a greater distance. For example, when the query sample is fault data, the LOF increases owing to the high weight assignment of distant neighbors.

3. When the distance values calculated in Equation (1) are the same, the number of neighbors may be greater than  $k$  defined in advance.  $reach - dist_k$  prevents the calculated distance from being greater than  $k$  and calculates the reachability distance of  $\mathbf{x}_{\text{query}}$ , as shown in Equation (4):

$$reach - dist_k(\mathbf{x}_{\text{query}}) = \max\{d_k(\mathbf{x}_{\text{query}}), d(\mathbf{x}_{\text{trn}}, \mathbf{x}_{\text{query}})\}, \tag{4}$$

where  $d_k(\mathbf{x}_{\text{query}})$  and  $d(\cdot)$  are the  $k$ -distance and Euclidean distance functions between  $\mathbf{x}_{\text{trn}}$  and  $\mathbf{x}_{\text{query}}$ , respectively.  $reach - dist_k$  means the reachability distance from  $\mathbf{x}_{\text{query}}$  to  $\mathbf{x}_{\text{trn}}$ , and the larger value among  $d_k(\mathbf{x}_{\text{query}})$  and  $d(\mathbf{x}_{\text{trn}}, \mathbf{x}_{\text{query}})$  is determined as  $reach - dist_k(\mathbf{x}_{\text{query}})$ .

4. The local reachability distance (lrd) is the reciprocal of the  $reach - dist_k(\mathbf{x}_{\text{query}})$  average and is calculated using Equation (5):

$$lrd_k(\mathbf{x}_{\text{query}}) = 1 / \left( \frac{\sum_{\mathbf{x}_{\text{query}} \in N_k(\mathbf{x}_{\text{query}})} reach - dist_k(\mathbf{x}_{\text{trn}}, \mathbf{x}_{\text{query}})}{k} \right) \tag{5}$$

To represent the degree of deviation of  $\mathbf{x}_{\text{query}}$  on a numerical scale, LOF is calculated using Equation (6):

$$LOF_k(\mathbf{x}_{\text{query}}) = \frac{\sum_{\mathbf{x}_{\text{query}} \in N_k(\mathbf{x}_{\text{query}})} \frac{lrd_k(\mathbf{x}_{\text{trn}})}{lrd_k(\mathbf{x}_{\text{query}})}}{|N_k(\mathbf{x}_{\text{query}})|} \tag{6}$$

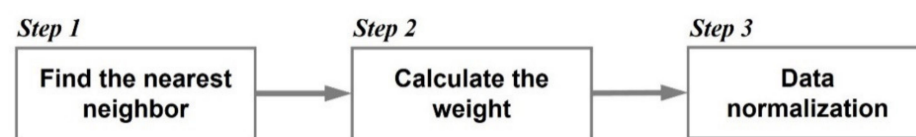
where LOF represents the average ratio of lrd, and when  $\mathbf{x}_{\text{query}}$  is similar to the training data, LOF is approximately close to one. For example, when  $\mathbf{x}_{\text{query}}$  is normal, LOF is close to one when the  $reach - dist_k(\mathbf{x}_{\text{query}})$  calculated in Equation (4) is small.

### 2.2. $k$ -Nearest Neighbor Normalization

In a multimode process, each mode follows a different distribution, such as Gaussian or non-Gaussian [38]. Therefore, various strategies for performing multimode fault detection have been proposed. Data normalization is an essential procedure for multimode fault detection. If fault detection is performed under conditions where the variance of each variable is different, erroneous results may be obtained. In general, the z-score normalization method is used to set the ranges of the variables to be the same and is calculated using Equation (7):

$$z = \frac{x - \mu}{\sigma}, \tag{7}$$

where  $\mu$  and  $\sigma$  represent the mean and standard deviation of the training data, respectively. The z-score method is effective when the data follow a single distribution. When each mode has a different distribution, this method uses the mean and standard deviation of the entire data; therefore, even after normalization, the mean and variance of each mode are still different [5]. In other words, the z-score method is not suitable for a process in which the operation mode changes because it uses a fixed global mean and variance for all the data. To avoid the shortcomings of the z-score method, we used  $k$ NN-based normalization.  $k$ NN, which normalizes data based on neighbors, is widely used in single-mode methods because it can eliminate multimode characteristics [1,5,6,34,39]. Guo et al. [18] confirmed that multimode characteristics can be removed by performing  $k$ NN-based local normalization. As shown in Figure 3, normalization using  $k$ NN is performed as follows: (1) neighbor selection through  $k$ NN; (2) weight allocation to surrounding neighbors; and (3) data normalization using the mean and weight of neighbors.



**Figure 3.** Procedure for  $k$ -nearest-based normalization.

- (1) We assume that the training data ( $\mathbf{X}_{\text{tm}} \in \mathbb{R}^{n \times m}$  with  $n$  samples and  $m$  variables) and a neighbor ( $N(\mathbf{x}_i)$ ) for each training sample are the same as in Equation (8):

$$N(\mathbf{x}_i) = \{\mathbf{x}_i^1, \mathbf{x}_i^2, \dots, \mathbf{x}_i^k\}, \text{ where } i = 1, 2, \dots, n, \quad (8)$$

where  $N(\mathbf{x}_i)$  represents  $k$  neighbors ( $k_{\text{kNS}}$ ) for a sample of the training data ( $\mathbf{x}_i$ ). The value of  $k_{\text{kNS}}$  should be set to be smaller than that of at least one single-mode subset among the multimode data.

- (2) The weight of  $\mathbf{x}_i$ , selected in Equation (8), is determined by Equation (9):

$$w_q = \frac{\frac{1}{d(\mathbf{x}_i, N(\mathbf{x}_i^q))}}{\sum_{s=1}^k \frac{1}{d(\mathbf{x}_i, N(\mathbf{x}_i^s))}}, \text{ where } q = 1, 2, \dots, k, \quad (9)$$

where  $w_q$  ( $q = 1, 2, \dots, k$ ) is the weight corresponding to each neighbor, and the sum of the weights is one.  $d(\mathbf{x}_i, N(\mathbf{x}_i^q))$  is the  $q$ th neighbor of  $\mathbf{x}_i$ . The weight calculated in Equation (9) represents a penalty based on the distance of the neighbor. For example, the smaller the calculated distance value between the query sample and neighbors, the greater the weight; thus, more information about the neighbor with a small distance is used.

- (3) Data is normalized as in Equation (10), using the mean of the neighbors of the  $\mathbf{x}_i$ , selected in Equation (8), and the weight of Equation (9):

$$z_i^{\text{knn}} = \mathbf{x}_i - \sum_{j=1}^k w_j \mathbf{m}_j, \quad (10)$$

where  $w_j$  and  $\mathbf{m}_j = [N_1(\mathbf{x}_i^q), N_2(\mathbf{x}_i^q), \dots, N_k(\mathbf{x}_i^q)]$  are the weights and averages of each neighbor, respectively. After calculating Equation (10), the mean and unit variance of the preprocessed data are zero, and approximately follow a uniform Gaussian distribution. The data ( $z_i^{\text{knn}}$ ) is preprocessed using the aforementioned process for all training data.

### 2.3. Setting Threshold Value by KDE

In this study, a non-parametric kernel density estimation (KDE) method was used to set a threshold value for fault detection. KDE, described by Rosenblatt et al. [40] and Parzen et al. [41], is a method to estimate a probability density function (PDF) from discrete samples. KDE estimates the PDF for all samples by summing the individual kernels of each sample [42,43]. Notably, the literature [44–46] verified that the threshold determined by KDE does not need to assume that the data follow a Gaussian distribution, which causes a malfunction. Therefore, they are widely used for fault detection. The estimation of the data distribution using the KDE is as follows: Given a univariate random variable, the PDF and cumulative distribution function (CDF) through the KDE are calculated using Equations (11) and (12):

$$\hat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^n K(x - x_i/h), \quad (11)$$

$$\hat{F}_h(x) = \frac{1}{n} \sum_{i=1}^n W(x - x_i/h), \quad (12)$$

where  $K(\cdot)$  is the kernel function,  $h$  is the smoothing parameter,  $W(t) = \int_{-\infty}^t K(u)du$ , and  $n$  is the number of samples. The KDE kernel function has various functions such as uniform, triangular, and Gaussian; however, the Gaussian kernel function is generally used. In this paper, the 'ksdensity' built in MATLAB's statistic and machine learning toolbox was used.

### 2.4. Detection Performance Indices for Fault Detection Validation

In this study, a confusion matrix was used as shown in Table 1 to evaluate the fault detection performance. Type I and II errors mean a false negative (FN) and false positive (FP), respectively. In the field of fault detection, two type errors are used because the ratio of normal to normal or fault to the fault is important. For example, if the type I error (FN) is low, the model results in fewer false alarms because the model judges normal to be normal. Conversely, when the type II error is high, the FP is high because the fault is judged to be normal. Additionally, precision ( $\frac{TP}{TP+FP}$ ), recall ( $\frac{TP}{TP+FN}$ ), F1 score ( $2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$ ), and accuracy ( $\frac{TP+TN}{TP+FN+FP+TN}$ ) are used to verify the detection performance of the fault detection methods.

**Table 1.** Confusion matrix for fault detection performance.

		Actual State	
		True	False
Decision	True	True Positive (TP)	False Positive (FP)
	False	False Negative (FN)	True Negative (TN)

## 3. Case Study

In this section, two types (drift and step error) of the multimode process are used to verify the fault detection performance of the proposed method and the case of an unplanned shutdown of a circulating fluidized bed combustion boiler is studied. First, we determine whether the proposed method can detect a fault more effectively than conventional methods through a multimode simulation process. Then, the applicability of the proposed method is confirmed by applying it to actual multimode cases. Section 3.1 compares the detection performance of the conventional methods (PCA, KPCA,  $k$ NN, LOF) and the proposed method for a numerical example with a multimode process. In a multimode process, the goal is to verify that artificially generated faults can be detected. Section 3.2 compares the detection performance of each method with that of a circulating fluidized bed combustion boiler. In this case, the goal is to detect a fault before a boiler stop occurs for a case that occurred in an actual boiler. The confidence level of KDE for setting the threshold was set at 0.01.

### 3.1. Multimode Numerical Example

A simple multimode numerical example was proposed by Ge et al. [3], and He et al. [5] redesigned the verification of the multimodal processes. This example consists of latent variables and five variables driven by sources, which can be generated from the following system:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 0.5768 & 0.3766 \\ 0.7382 & 0.0566 \\ 0.8291 & 0.4009 \\ 0.6519 & 0.2070 \\ 0.3972 & 0.8045 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \end{bmatrix}, \tag{13}$$

where  $[e_1, e_2, \dots, e_5]^T$  is the white noise with zero-mean and a standard deviation of 0.01.  $[s_1, s_2]^T$  is a variable that represents three different operating modes, and the mean and variance of each mode are expressed by Equation (14):

$$\begin{aligned} \text{Mode 1 : } & s_1 \sim N(10, 0.8) \\ & s_2 \sim N(12, 1.3) \\ \text{Mode 2 : } & s_1 \sim N(5, 0.6) \\ & s_2 \sim N(20, 0.7) \\ \text{Mode 3 : } & s_1 \sim N(16, 1.5) \\ & s_2 \sim N(30, 2.5) \end{aligned} \tag{14}$$

The amount of training data generated using Equations (13) and (14) was 1200 and 400, respectively, for each mode. There are 200 query samples: from 1 to 100 are normal, and 101, where drift- or step-type faults occur. The generated training data and query samples are shown in Figure 4. The training data consist of a total of three modes, and the mean and variance according to each mode are the same as those in Equation (14).

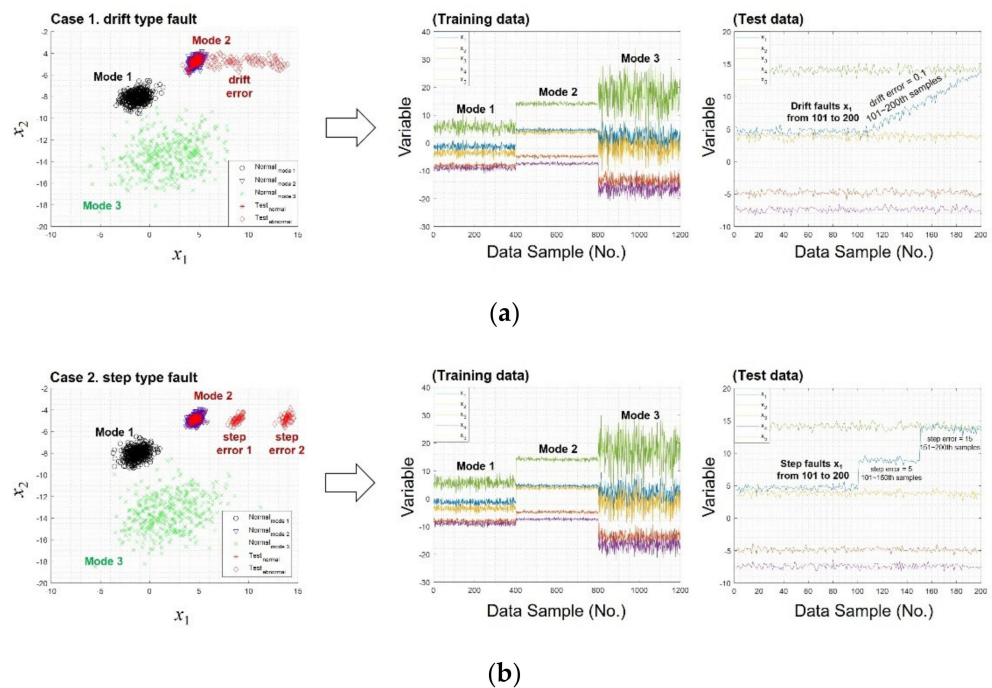


Figure 4. Generated multimode data: (a) drift type fault case; (b) step type fault case.

- Case 1. The system was initially running in mode 2, and then, a drift error of 0.1 was added to  $x_1$  from the 101st through to the 200th samples.



- Case 2. The system was initially running in mode 2, with 5 ( $k-100$ ) applied to the 101st to 150th samples with a bias error of 5, where  $k$  denotes the serial number of the test sample, and a bias error of 15 from the 151st to 200th samples was applied.

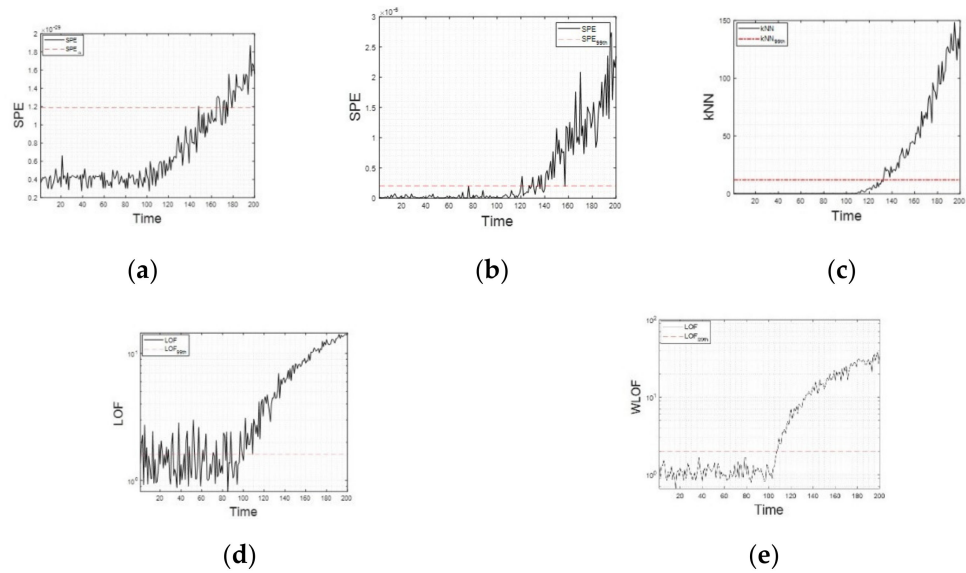
Two modes (mode 1 and mode 2) of the training data were generated with dense data, and mode three was generated with sparse density. Figure 4a is a drift-type fault (case 1) that occurred in mode two, and the size of the fault increased by 0.1 (from 0.05 to 10) from 101 to 200. Figure 4b shows a step-type fault (case 2) that occurred in mode two, and the bias error increased by 5 from 101 to 150 and increased by 15 from 151 to 200. Specifically, a step fault corresponds to the case in which a two-step fault occurs.

In the mode data, the average of each variable is different by  $\pm 5$  or more because it operates in different modes as shown in Figure 4. In the case of mode three, the variance is larger than in modes one and two, and the density between the data is relatively sparse, resulting in large fluctuations in the data. In addition, it can be confirmed that the data has a mixed distribution while various modes are included in the training data. Due to the characteristics of these various modes, the average of modes two and three is significantly different compared to mode one, and thus it may be determined as a fault. To detect multimode faults, the threshold values of PCA, KPCA,  $k$ NN, LOF, and the proposed method were set using KDE ( $\alpha = 0.01$ ). In the case of PCA and KPCA, SPE, which showed a higher detection performance among  $T^2$  statistics and SPE, was used. The number of neighbors of the conventional  $k$ NN, LOF, and WLOF was 15.

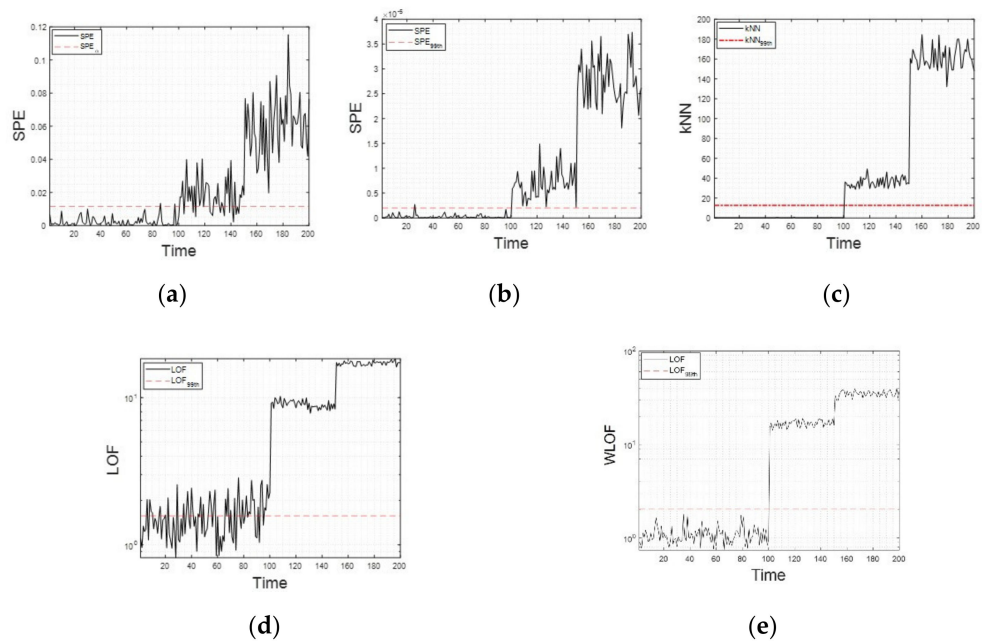
Figure 5 shows the fault detection results of the PCA, KPCA,  $k$ NN, LOF, and  $k$ NS-WLOF for case 1. In PCA, three modes were included in the training data; thus, the threshold was set high compared to LOF and  $k$ NS-WLOF. Therefore, the fault detection time ( $t = 160$ ) of PCA was delayed. This shows that PCA cannot effectively detect faults occurring in the multimode process because the detection performance deteriorates in the multimode process in which several distributions are mixed owing to the assumption of the distribution of specific data. Although the performance of KPCA was improved compared to PCA, the detection time was delayed, resulting in high type II errors. In contrast to PCA and KPCA,  $k$ NN and LOF is a distance-based method that can detect a fault even when the distribution of the training data is mixed. Therefore, it is shown that the conventional  $k$ NN, LOF, and proposed method can detect drift-type faults. However, in two conventional methods, many query samples are out of the threshold in the normal section; therefore, the type I error is high. PCA, KPCA,  $k$ NN, and LOF were normalized using the z-score method, such that multimodal characteristics remained in the normal data. Consequently, it is difficult to perform accurate fault detection beyond the threshold value despite such a normal section. In contrast,  $k$ NS-WLOF shows that the remaining fault can be effectively detected except for some sections where the initial faults start to occur, and the fault size is small. In particular, even in the normal section,  $k$ NS-WLOF is calculated to be lower than the threshold value to prevent type I errors. The performance of the proposed method is superior to the conventional PCA and LOF because if the distance between the normal data and the recognized neighbor is less than  $\frac{2}{k}$ , the distance is adjusted by dividing the distance by a weight.

Figure 6 shows the fault detection results of conventional PCA, KPCA,  $k$ NN, and LOF, and  $k$ NS-WLOF for case 2. As shown in Figure 6a, PCA has higher variability according to the fault size compared to conventional LOF and  $k$ NS-WLOF, and some sections ( $t = 101 \sim 150$ ) were calculated to be smaller than the threshold. Therefore, PCA cannot properly detect drift or a step-type fault. Although KPCA has an improved performance over PCA, it still shows high variability. In conventional  $k$ NN, the fault of the step error was effectively detected, unlike the drift error. In the conventional LOF, the fault was effectively detected for the step fault ( $t = 151 \sim 200$ , bias error = 15), although many false alarms occurred in the normal section. This is because the multimode characteristic remains the same as in case 1; thus, the fault section was considered normal. Therefore, conventional PCA and LOF cause confusion among operators owing to many false alarms or erroneous diagnosis; it is difficult to determine whether they are accurate fault detection

models. Compared to the conventional PCA and LOF, KPCA shows that faults caused by bias error can be detected. However, it shows that the threshold value is set high because other mode data is included in the training data.



**Figure 5.** Comparison of drift-type fault detection results: (a) conventional PCA; (b) conventional KPCA; (c) conventional  $k$ NN; (d) conventional LOF; (e)  $k$ NS–WLOF.



**Figure 6.** Comparison of drift-type fault detection results: (a) conventional PCA; (b) conventional KPCA; (c) conventional  $k$ NN; (d) conventional LOF; (e)  $k$ NS–WLOF.

In contrast to the conventional methods (PCA,  $k$ NN, LOF), the proposed method does not deviate from the threshold value in the normal section; therefore, it can be confirmed that the case 2 fault is accurately detected while lowering the false alarm. It can be validated that the  $k$ NS–WLOF is lower than the threshold value in the normal section compared to the conventional LOF. In summary, we compared the detection performance with two multimode process numerical examples and showed that the proposed method can effectively detect faults with fewer false alarms than the conventional methods.

The results of the detection performance for two multimode cases are shown in Table 2. In conventional PCA, the normal section ( $t = 1 \sim 100$ ) was not considered a fault; thus, a type I error did not occur. However, a high type II error was calculated because the initial fault after both types (drift and step error) was not detected. The reason is that the threshold value for fault detection is set relatively high because the training data with the multimode characteristics are used. In KPCA, detection delay occurred, and drift error resulted in high type II error, but faults of step error were properly detected.  $k$ NN increased type II error due to detection delay in drift error, but the step error effectively detected the fault as in KPCA. Conventional LOF effectively detected two types of faults compared with PCA but determined that the normal section ( $t = 1 \sim 100$ ) was a fault, resulting in a high type I error. However, the proposed method effectively detected the fault without causing a type I error in the normal section. In particular, PCA, KPCA, and  $k$ NN were calculated with low precision because the initial defect was considered normal. Thus, even if the recall is high, the F1 score is low due to low precision. In the case of conventional LOF, the precision was high, but the F1 score was calculated to be low because the recall was low due to the low number of type I errors. Compared with the conventional methods, the proposed method has a low number and FP; thus, precision, recall, F1 score, and accuracy were all calculated to be high. In summary, a comparison of the detection performance using two multimode process numerical examples shows that the proposed method can more effectively detect the fault than the other two conventional methods.

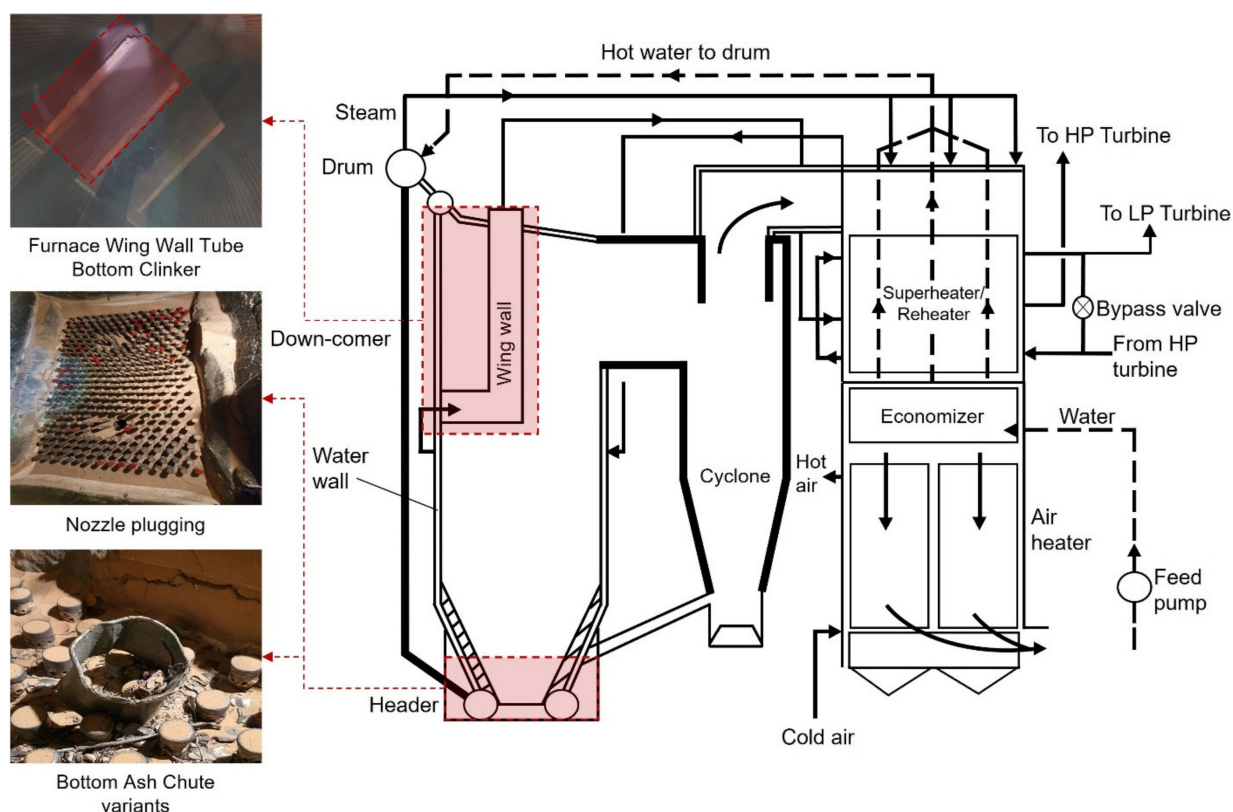
**Table 2.** Performance indices of the proposed method and comparison methods.

		TN (Type I)	FP (Type II)	Precision (%)	Recall (%)	F1 Score (%)	Accuracy (%)
PCA	Case 1	<b>0</b>	69	59.17	<b>100</b>	74.35	65.5
(SPE)	Case 2	2	16	85.96	98	91.59	91
KPCA	Case 1	<b>0</b>	35	74.07	<b>100</b>	85.11	82.5
(SPE)	Case 2	<b>0</b>	<b>0</b>	<b>100</b>	99	99.5	99.5
$k$ NN	Case 1	<b>0</b>	32	75.76	<b>100</b>	86.21	84
( $D^2$ )	Case 2	<b>0</b>	<b>0</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>
LOF	Case 1	33	<b>0</b>	<b>100</b>	67	80.24	83.5
(LOF)	Case 2	42	<b>0</b>	<b>100</b>	58	73.42	79
$k$ NS-WLOF	Case 1	<b>0</b>	7	93.45	<b>100</b>	<b>96.62</b>	<b>96.5</b>
(LOF)	Case 2	<b>0</b>	<b>0</b>	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>

The entries corresponding to the lowest type I and II errors are indicated in bold face.

### 3.2. Circulating Fluidized Bed Combustion Boiler (CFBC)

Circulating fluidized bed combustion boilers (CFBCs) are widely used in cogeneration power plants and small- and medium-sized power plants because they can generate electricity using low-quality fuels such as biomass (e.g., methane, ethanol, and hydrogen) and solid municipal waste fuels (e.g., wood, agricultural by-products, etc.). The power generation process of the CFBC burning fuel under the gas-solid flow condition inside the combustion furnace is shown in Figure 7. At this time, the heated high-temperature layer material scatters and circulates, and heat is transferred to the heat transfer tube to generate steam. The steam encounters the superheater again, and the steam generated in the heat exchanger is converted to high-temperature steam and fed to the turbine for power generation. Such power generation has the advantage of reducing the amount of waste to be landfilled and the use of fossil fuels.



**Figure 7.** Distribution diagram of CFBC and clinker and nozzle plugging of three furnace [47].

However, because CFBC uses bed materials such as sand, the alkali salt contained in the exhaust gas is attached to the bed material and the heat pipe in the boiler, causing erosion, corrosion, and agglomeration. For example, as potassium salts (KCl) contained in the fuel are converted to the gas phase, they react with chromium (Cr) and  $\text{Cr}_2\text{O}_3$  oxide films in a boiler tube to generate  $\text{K}_2\text{CrO}_4$  and  $\text{Cl}(\text{g})$  [48]. NaCl and KCl mainly coagulate in incineration boilers using domestic waste as fuel [49]. The agglomeration phenomenon is mainly caused by the alkaline components potassium (K) and sodium (Na) in waste fuel, and occurs as follows: low-melting-temperature silicate is formed together with the silica component ( $\text{SiO}_2$ ) of the sand; (1) ash is melted and the surface of the bed material is coated with stickiness and gradually begins to be deposited. (2) In a high-temperature environment, the deposited ash particles melt and aggregate. (3) The low boiling point of the volatilized alkali metal condenses, it reacts, the melting point is lowered, and agglomeration occurs.

In general, food waste and steel materials among domestic wastes are selected and separated in advance. However, it is possible that some biomass components are contained in solid fuel and are introduced into the power generation facilities. The agglomeration phenomenon mainly occurs around the combustion and cyclone and reduces the heat-transfer efficiency of the boiler by inducing slagging, clinker, and fouling phenomena. If the appropriate actions are not performed after agglomeration in the fluidized bed, it will eventually lead to an unplanned shutdown. In summary, CFBC is caused by the inevitability of the combustion of various fuels and oxidizers to generate ashes, which are by-products of combustion. This leads to problems such as de-fluidization, the formation of deposits in the cyclone, and plugging in the recirculation area [50,51]. Sediment formation can be prevented by controlling the temperature of the fluidized bed furnace [52].

In this study, we compared the detection performance of the proposed method and two conventional methods in the case of unplanned shutdown owing to clinker and nozzle plugging in the furnace in CFBC. The generator capacity of the CFBC was 9.1 MW. The maximum capacity, maximum inlet steam pressure, maximum steam temperature,

and maximum steam pressure of the steam turbine were 72 t/h, 41 ata, 420, and 0.3 ata, respectively. This shutdown was stopped at 10 a.m. on 8 June 2020, owing to a sudden increase in the differential pressure between the furnace and the boiler and the differential pressure between the super heater (S/H) and the reheater (R/H). As shown in Table 3, 113 boiler- and steam-related variables collected at 10-s (10,000 m/s) intervals through the supervisory control and data acquisition (SCADA) system were used for fault detection. A total of 113 CFBC variables were selected by power plant experts as major power plant-related variables for fault detection. For the training and testing data, 70,000 (25 May 2020–2 June 2020) and 32,768 (5 June 2020–8 June 2020) data were used. The training data were composed of data including several operational modes operated during the operation of the power plant. Figure 7 shows a diagram of the CFBC. CFBC has problems such as layer material agglomeration, deposition generation, and nozzle plugging owing to the accumulation of precipitates at the lower and upper part of the furnace owing to the bed material. As Figure 7 shows, the fault case also caused more than 40 nozzles to plug at the lower end of the furnace and clinker at the lower end of the wing wall tube owing to sediment.

**Table 3.** Summary of monitored variables for CFBC.

No.	Description	Unit	No.	Description	Unit	No.	Description	Unit
$x_1$	steam output of feedwater pipe 1 (sensor A)	t/h	$x_{39}$	inlet temp. of feedwater pipe 1 (sensor B)	°C	$x_{77}$	inlet output of feedwater pipe 1	%
$x_2$	steam output of feedwater pipe 1 (sensor B)	t/h	$x_{40}$	inlet temp. of feedwater pipe 2 (sensor A)	°C	$x_{78}$	outlet output of feedwater pipe 1	%
$x_3$	steam output of feedwater pipe 2 (sensor C)	t/h	$x_{41}$	inlet temp. of feedwater pipe 2 (sensor B)	°C	$x_{79}$	outlet output of feedwater pipe 2	%
$x_4$	steam output of fluidized bed material supply	t/h	$x_{42}$	outlet temp. of feedwater pipe 1	°C	$x_{80}$	output of feedwater ratio (sensor A)	%
$x_5$	aux steam output of lower feedwater pipe	t/h	$x_{43}$	outlet temp. of feedwater pipe 2	°C	$x_{81}$	output of feedwater ratio (sensor B)	%
$x_6$	steam flow of feedwater pipe 1	t/h	$x_{44}$	inlet temp. of fluidized bed material supply	°C	$x_{82}$	output of steam ratio (sensor A)	%
$x_7$	steam flow of feedwater pipe 2	t/h	$x_{45}$	inlet temp. of lower place furnace (sensor A)	°C	$x_{83}$	output of steam ratio (sensor B)	%
$x_8$	steam flow of fluidized bed material supply	t/h	$x_{46}$	inlet temp. of lower place furnace (sensor B)	°C	$x_{84}$	output of steam ratio (sensor C)	%
$x_9$	furnace press. of feedwater pipe 2	mm H <sub>2</sub> O	$x_{47}$	inlet temp. of lower place furnace (sensor C)	°C	$x_{85}$	amount of H <sub>2</sub> O	%
$x_{10}$	furnace press. of feedwater pipe 2 (sensor A)	mm H <sub>2</sub> O	$x_{48}$	inlet temp. of middle place furnace (sensor A)	°C	$x_{86}$	inlet press. of feedwater pipe 2	mm H <sub>2</sub> O
$x_{11}$	furnace press. of feedwater pipe (sensor B)	mm H <sub>2</sub> O	$x_{49}$	inlet temp. of middle place furnace (sensor B)	°C	$x_{87}$	diff. press. outlet between feedwater pipe 2	mm H <sub>2</sub> O
$x_{12}$	combustor bed press. of lower furnace feedwater (sensor A)	mm H <sub>2</sub> O	$x_{50}$	outlet temp. of lower place furnace	°C	$x_{88}$	steam flow of air pre-heater and dry reactor	mm H <sub>2</sub> O
$x_{13}$	combustor bed press. of lower furnace feedwater (sensor B)	mm H <sub>2</sub> O	$x_{51}$	inlet temp. of upper place furnace	°C	$x_{89}$	diff. of press. between furnace and top of cyclone	mm H <sub>2</sub> O
$x_{14}$	sum of steam output of feedwater pipe 1 and 2	mm H <sub>2</sub> O	$x_{52}$	outlet temp. of upper place furnace	°C	$x_{90}$	diff. of press. 2nd and 1st S/H.	mm H <sub>2</sub> O
$x_{15}$	press. of fluidized bed material supply	mm H <sub>2</sub> O	$x_{53}$	inlet temp. of furnace	°C	$x_{91}$	diff. of press. 1st S/H and 2nd eco.	mm H <sub>2</sub> O
$x_{16}$	press. of lower place furnace	mm H <sub>2</sub> O	$x_{54}$	inlet temp. of cyclone and boiler front-end	°C	$x_{92}$	diff. of press. 2nd and 1st eco.	mm H <sub>2</sub> O
$x_{17}$	press. of middle place furnace	mm H <sub>2</sub> O	$x_{55}$	inlet temp. of cyclone and boiler terminal	°C	$x_{93}$	diff. of press. of 1st and new eco.	mm H <sub>2</sub> O
$x_{18}$	press. of upper place furnace	mm H <sub>2</sub> O	$x_{56}$	inlet temp. of cyclone and boiler middle point	°C	$x_{94}$	diff. of press. of new eco.	mm H <sub>2</sub> O
$x_{19}$	press. between cyclone and boiler	mm H <sub>2</sub> O	$x_{57}$	inlet temp. of 1st S/H	°C	$x_{95}$	metering bin A outlet conveyor	rpm

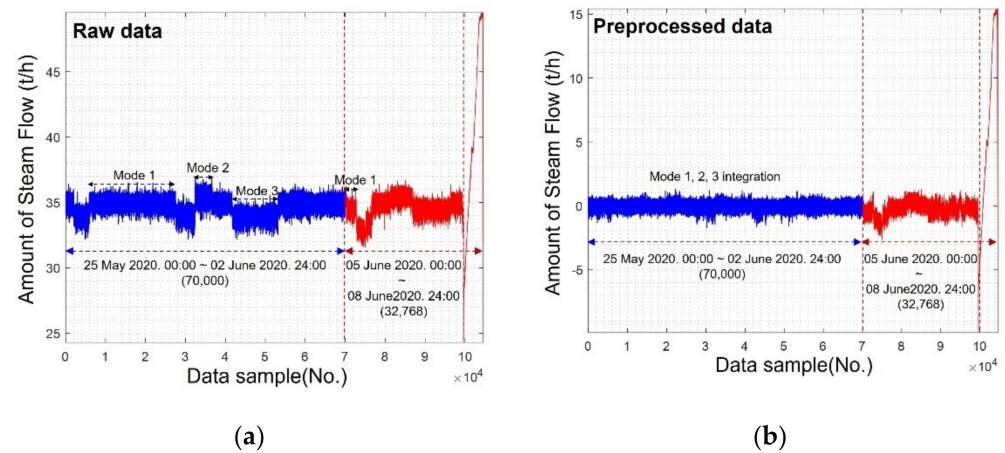
Table 3. Cont.

No.	Description	Unit	No.	Description	Unit	No.	Description	Unit
$x_{20}$	press. of 1st S/H	mm H <sub>2</sub> O	$x_{58}$	inlet temp. of 2nd S/H	°C	$x_{96}$	outlet temp. of 1st S/H	°C
$x_{21}$	press. of 2nd S/H	mm H <sub>2</sub> O	$x_{59}$	inlet temp. of 1st eco.	°C	$x_{97}$	outlet temp. of 1st S/H	°C
$x_{22}$	press. of steam supplied of upper place furnace	MPa	$x_{60}$	inlet temp. of 2nd eco.	°C	$x_{98}$	inlet temp. of 2nd S/H (sensor A)	°C
$x_{23}$	press. of 2nd eco.	mm H <sub>2</sub> O	$x_{61}$	outlet temp. of upper place boiler	°C	$x_{99}$	inlet temp. of 2nd S/H (sensor B)	°C
$x_{24}$	press. of lower supply cyclone (sensor A)	mm H <sub>2</sub> O	$x_{62}$	inlet temp. of cyclone fluidized bed material supply	°C	$x_{100}$	temp. of steam supplied of boiler silencer	°C
$x_{25}$	press. of lower supply cyclone (sensor B)	mm H <sub>2</sub> O	$x_{63}$	inlet temp. of dry reactor and bag filter	°C	$x_{101}$	inlet temp. of 1st S/H (sensor A)	°C
$x_{26}$	press. of middle place cyclone	mm H <sub>2</sub> O	$x_{64}$	inlet temp. of SCR and SGR	°C	$x_{102}$	inlet temp. of 1st S/H (sensor B)	°C
$x_{27}$	press. of middle place furnace	mm H <sub>2</sub> O	$x_{65}$	inlet temp. of SGR and combustor	°C	$x_{103}$	steam drum level of feedwater tank	rpm
$x_{28}$	press. of lower place furnace	mm H <sub>2</sub> O	$x_{66}$	inlet temp. of feedwater pipe 1	°C	$x_{104}$	outlet press. 2nd S/H	MPa
$x_{29}$	steam press. of SCR	mm H <sub>2</sub> O	$x_{67}$	inlet temp. of feedwater pipe 2	°C	$x_{105}$	outlet press. steam supplied of 2nd S/H	MPa
$x_{30}$	press. of air pre-heater	mm H <sub>2</sub> O	$x_{68}$	outlet temp. of dry reactor front-end	°C	$x_{106}$	inlet press. 2nd S/H	MPa
$x_{31}$	press. of air pre-heater and dry reactor 1	mm H <sub>2</sub> O	$x_{69}$	outlet temp. of air pre-heater terminal	°C	$x_{107}$	amount of outlet steam flow 2nd S/H	t/h
$x_{32}$	press. of dry reactor and bag filter	mm H <sub>2</sub> O	$x_{70}$	diff. of temp. 2nd and 1st S/H	°C	$x_{108}$	amount of steam flow 2nd S/H	t/h
$x_{33}$	diff. press. between dry reactor and bag filter	mm H <sub>2</sub> O	$x_{71}$	diff. of temp. 1st S/H and 2nd eco.	°C	$x_{109}$	steam output of steam drum	t/h
$x_{34}$	press of upper place combustor	mm H <sub>2</sub> O	$x_{72}$	diff. of temp. 2nd and 1st eco.	°C	$x_{110}$	steam drum level of eco.	t/h
$x_{35}$	press. of SCR terminal	mm H <sub>2</sub> O	$x_{73}$	diff. of temp. 1st S/H and new eco.	°C	$x_{111}$	outlet press. of 2nd S/H 1-1	MPa
$x_{36}$	diff. press. between feedwater pipe 1	mm H <sub>2</sub> O	$x_{74}$	diff. of temp. new eco. and bag filter	°C	$x_{112}$	outlet press. of S/H 1-2	MPa
$x_{37}$	diff. press. of feedwater pipe 1 (sensor A and B)	mm H <sub>2</sub> O	$x_{75}$	diff. of temp. cyclone and boiler	°C	$x_{113}$	output of steam drum	%
$x_{38}$	inlet temp. of feedwater pipe 1 (sensor A)	°C	$x_{76}$	amount of O <sub>2</sub> in eco.	%			

Figure 8 shows the training data, including the multiple modes and test data before the fault. Figure 8a shows three modes with different means and variances. As shown in Figure 8a, it is difficult to determine the correct mode after the test data are operated in mode 1 and a change similar to mode three occurs. Therefore, the test data were considered to have failed during the change from mode 1 to another mode. If fault detection is performed without classifying or integrating the data according to each mode, erroneous diagnosis occurs. To handle this challenge, three-mode data were normalized to data following a single distribution using the *k*NS strategy as shown in Figure 8b. Because the mean and variance of each mode of the normalized data are the same, fault detection can be performed using a single model.

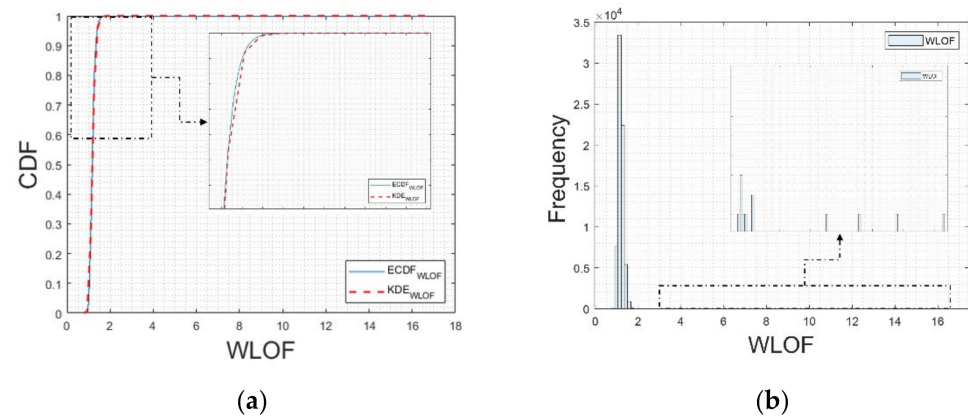
Compared with conventional PCA, KPCA, *k*NN, and LOF, the proposed method can effectively detect a fault operating in various modes. In Figure 8a, when only a single model is used under the multimode operation condition, miss detection results occur. Unfortunately, PCA has a limited performance on simulated data and real multimode processes because it requires assumptions about the data distribution. KPCA requires a lot of time for model training, and it is difficult to determine the kernel functions and parameters for fault detection. Conventional LOF are far apart from each other even for normal data because the mean or variance varies between operation modes. For this reason, conventional methods are limited to being applied to a multimode process. On the other hand, the proposed method is normalized to an integrated mode using the local

normalization strategy and then detects a fault. Therefore, the proposed method can detect a fault in complex industrial processes operating in various modes.



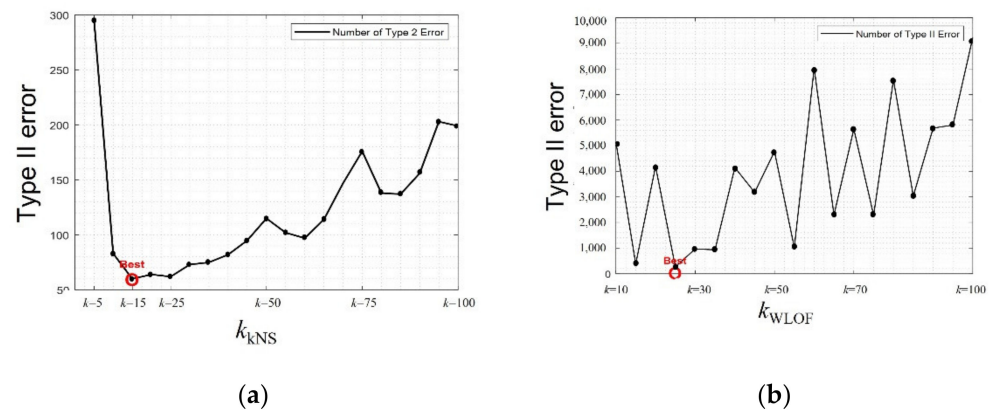
**Figure 8.** Training and test data for CFBC fault detection: (a) raw CFBC data; (b) preprocessed CFBC data.

Figure 9 shows the histogram of the WLOF calculated using training data and CDFs estimated using KDE to perform fault detection. To estimate the probability distribution of WLOF, the ‘ksdensity’ function built into MATLAB was used, and  $\alpha$  of the threshold was set to 0.01. In Figure 9, the empirical CDF and estimated CDF via KDE are indicated by blue and dashed red lines, respectively. As shown in Figure 9, the empirical CDF and distribution estimated through the KDE are similar.



**Figure 9.** Histogram and empirical and estimated CDFs of WLOF from the training data: (a) histogram; (b) empirical CDF and estimated CDF via KDE.

Figure 10 shows the results of selecting the optimal parameters ( $k_{kNS}$ ,  $k_{WLOF}$ ) used in the proposed method for CFBC fault detection. As shown in Figure 10a, data normalization was performed while gradually increasing the number of neighbors used for data normalization, and the results of the type II error of the WLOF are provided. When  $k_{kNS}$  was 15, the type II error was the minimum, and the error continued to increase thereafter. Figure 10b shows the type II error according to the number of neighbors ( $k_{WLOF}$ ) of the WLOF. This shows that the error trend gradually increased after  $k_{WLOF}$  reached 30. Through the aforementioned experiment, the  $k_{NS}$  normalized neighbors and WLOF neighbors of the proposed method were set to 15 and 25, respectively.



**Figure 10.** Results of the  $k_{NS}$  parameter ( $k_{kNS}$ ) and WLOF ( $k_{WLOF}$ ) selection using cross-validation: (a)  $kNS$  results; (b) WLOF results.

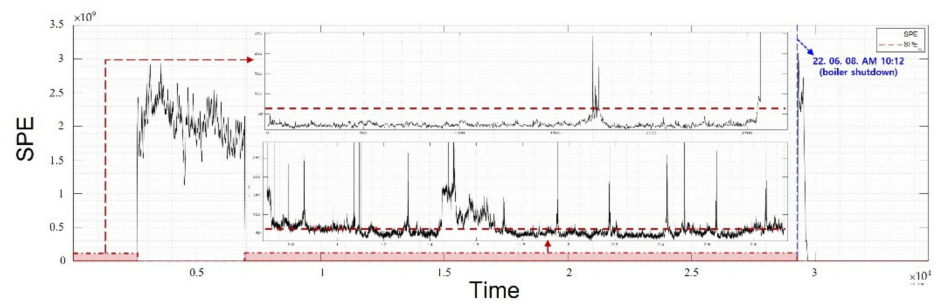
Figure 11 shows the detection performance of CFBC for PCA, KPCA,  $kNN$ , LOF, and  $kNS$ -WLOF. In the case of PCA and KPCA, SPE, which showed a higher detection performance among the  $T^2$  statistics and SPE, was used. In Figure 11a–d, the conventional PCA, KPCA,  $kNN$ , and LOF capture only the moment when mode 1 is changed to other modes and cannot detect abnormal actions. In particular, it is difficult to determine whether a fault has been detected because the value (SPE,  $D^2$ , LOF) drops below the threshold again after a significant bounce in mode one. The conventional PCA, KPCA,  $kNN$ , and LOF do not detect the fault because the characteristics between each mode remain after z-score normalization. Additionally, the fault data are similar to other modes of normal data, and the size of the fault is small compared to that before the boiler shutdown occurs, rendering it difficult to detect. However, as shown in Figure 11, the proposed method normalizes the data according to each mode to follow a single distribution through the  $kNS$  strategy; thus, fault detection is possible through a single model.

Furthermore, in contrast to conventional LOF, WLOF provides a large penalty when there are few neighbors to normal data with fault data, allowing the distance between the normal data and distant neighbors to be readjusted. Specifically, the WLOF can effectively detect a fault even if the distance between the normal and fault data is small.  $kNS$ -WLOF shows that the WLOF value continuously exceeds the threshold from the period in which the value is significantly increased until the boiler is stopped. Therefore, it was confirmed that an unplanned shutdown can be prevented by detecting the fault at an early stage using the proposed method.

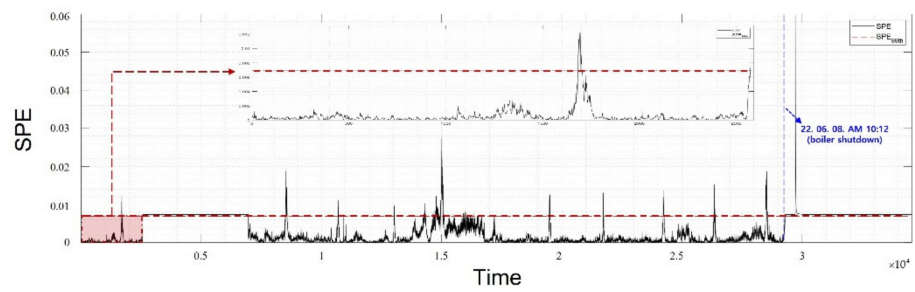
Table 4 summarizes the results of the detection performance (type I and II errors, precision, recall, F1 score, accuracy) in the fault case of the CFBC between PCA, KPCA,  $kNN$ , LOF, and the proposed method. In PCA and KPCA, the SPE value was smaller than the threshold value in the normal section; therefore, fewer false alarms occurred. However, PCA and KPCA could not capture the abnormal symptoms that occurred after changing from mode one to another mode, resulting in a high type II error (PCA:42.59, KPCA:66.73). In  $kNN$ , the type I error was low, but the fault data was judged as normal, and the type II error was high. In the conventional LOF, many false alarms occurred in the normal section; therefore, type I error was calculated to be high. Conventional methods had low recall due to low TN, but they were calculated with high precision because there were many cases where the fault section was detected as normal. Thus, the F1 score, which is the harmonic mean of precision and recall, is low. Consequently, the conventional PCA, KPCA,  $kNN$ , and LOF show that the fault occurring in the CFBC cannot be properly detected. The proposed method shows that the type I and II errors are lower than those of both methods, as shown in Figure 11, and abnormal symptoms can be detected before a fault occurs. WLOF correctly detected normal and fault data, and the precision was over 90%. In addition, the F1 score and accuracy was more than 50% and 20% higher than that of the conventional methods,



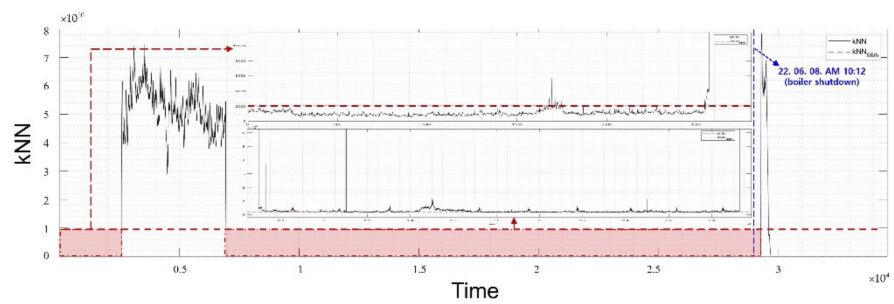
respectively. Therefore, it was confirmed that the proposed method can effectively detect a fault compared with conventional methods.



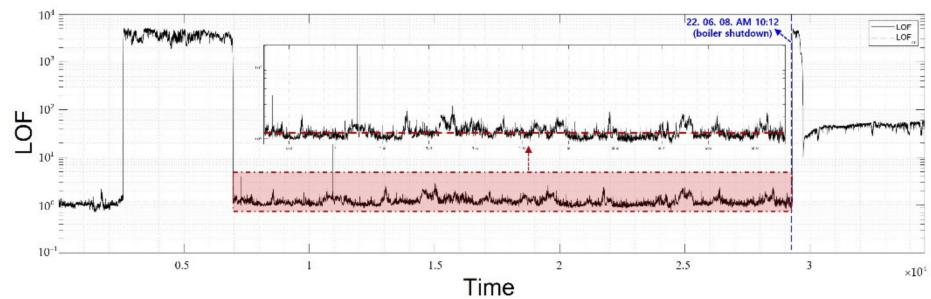
(a)



(b)

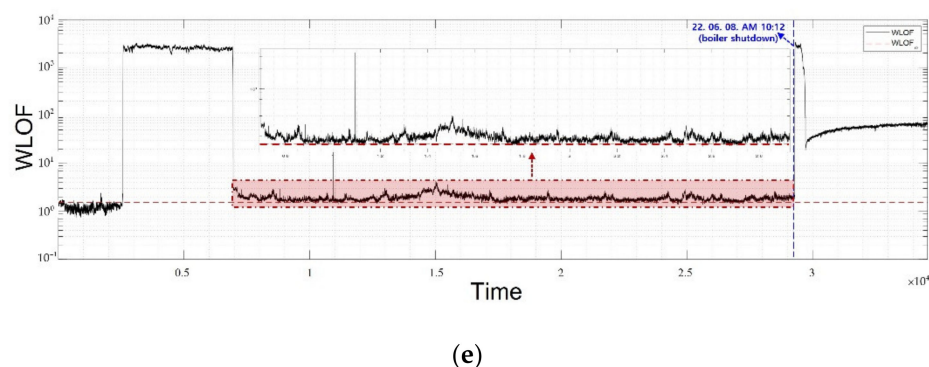


(c)



(d)

Figure 11. Cont.



**Figure 11.** Comparison of the fault detection results: (a) conventional PCA; (b) conventional KPCA; (c)  $k$ NN; (d) conventional LOF; (e)  $k$ NS–WLOF.

**Table 4.** Performance indices of the proposed method and comparison methods.

	TN (Type I)	FP (Type II)	Precision (%)	Recall (%)	F1 Score (%)	Accuracy (%)
PCA (SPE)	<b>0.78</b>	45.98	14.69	<b>99.22</b>	25.59	57.36
KPCA (SPE)	0.86	68.04	10.41	99.14	18.85	36.92
$k$ NN ( $D^2$ )	1.45	22.58	25.82	98.55	40.92	78.98
LOF (LOF)	10.57	38.89	15.50	89.43	26.42	63.20
$k$ NS- WLOF (LOF)	5.12	<b>0.78</b>	<b>90.68</b>	94.87	<b>92.73</b>	<b>98.90</b>

The entries corresponding to the lowest type I and II errors are indicated in bold.

#### 4. Conclusions

In this study, a novel fault detection method called  $k$ NS-WLOF is proposed for effectively detecting multimode processes. First, local normalization was performed based on neighbors using  $k$ NN instead of the z-score method to remove multimodal features. Local normalization was performed, and WLOF was used to detect faults measured in real time. To verify the effectiveness of the proposed method, the detection performances of the conventional methods and the proposed method were compared in the multimodal numerical case and CFBC unplanned shutdown case. The experimental results confirm that the proposed method can detect faults more effectively than the conventional PCA, KPCA, and LOF. In particular, the conventional methods using the z-score method in multimode cases and CFBC fault cases were erroneously determined as faults because they failed to remove the multimodal characteristics of the data for each mode. Because the proposed method removes multimodal features using  $k$ NN-based local normalization, the variables of the other modes included in the training data are in the same range. In addition, the proposed method has a lower error than conventional PCA and LOF because it is readjusted to the penalized distance value when there are many distant neighbors. In summary, it is confirmed that the two conventional methods have difficulty detecting a fault in a multimode process, whereas the proposed method can adequately detect a fault. Therefore, the proposed method can be applied to a real process operating in multiple modes.

In future research, we will consider the following two topics. First, in this study, the distance readjustment of the WLOF was empirically set as a penalty criterion because weights were allocated according to the number of  $k_{WLOF}$ . To address this issue, it will be set to an appropriate value using various cross-validation methods such as grid search and leave-one-out. Second, the LOF computation time increases according to the amount of data or the number of variables. For complex processes such as circulating fluidized bed boilers, the use of huge amounts of data collected in real time takes a lot of time to calculate the WLOF, which ultimately delays monitoring alarms. If the detection time is delayed, the

fault may spread to other equipment and lead to unplanned shutdown. Therefore, before applying WLOF, we will consider a study to remove unnecessary data using dimensionality reduction techniques such as PCA and ICA.

**Author Contributions:** S.J. and B.K. conceived and designed the simulations; J.K. (Jinyong Kim) and E.K. analyzed the data; J.K. (Jonggeun Kim) advised on the whole process of manuscript preparation; M.K. analyzed the data and wrote the paper. The analysis results and the paper were supervised by S.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C2009667).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

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

## References

1. Wang, G.; Liu, J.; Zhang, Y.; Li, Y. A novel multi-mode data processing method and its application in industrial process monitoring. *J. Chemom.* **2015**, *29*, 126–138. [[CrossRef](#)]
2. Yu, J.; Qin, S.J. Multimode process monitoring with Bayesian inference-based finite Gaussian mixture models. *AIChE J.* **2008**, *54*, 1811–1829. [[CrossRef](#)]
3. Ge, Z.; Song, Z. Multimode process monitoring based on Bayesian method. *J. Chemom. A J. Chemom. Soc.* **2009**, *23*, 636–650. [[CrossRef](#)]
4. Tan, S.; Wang, F.; Peng, J.; Chang, Y.; Wang, S. Multimode process monitoring based on mode identification. *Ind. Eng. Chem. Res.* **2012**, *51*, 374–388. [[CrossRef](#)]
5. Ma, H.; Hu, Y.; Shi, H. A novel local neighborhood standardization strategy and its application in fault detection of multimode processes. *Chemom. Intell. Lab. Syst.* **2012**, *118*, 287–300. [[CrossRef](#)]
6. Guo, J.; Yuan, T.; Li, Y. Fault detection of multimode process based on local neighbor normalized matrix. *Chemom. Intell. Lab. Syst.* **2016**, *154*, 162–175. [[CrossRef](#)]
7. Ge, Z.; Gao, F.; Song, Z. Two-dimensional Bayesian monitoring method for nonlinear multimode processes. *Chem. Eng. Sci.* **2011**, *66*, 5173–5183. [[CrossRef](#)]
8. Yu, J. A nonlinear kernel Gaussian mixture model based inferential monitoring approach for fault detection and diagnosis of chemical processes. *Chem. Eng. Sci.* **2012**, *68*, 506–519. [[CrossRef](#)]
9. Harrou, F.; Saidi, A.; Sun, Y.; Khadraoui, S. Monitoring of photovoltaic systems using improved kernel-based learning schemes. *IEEE J. Photovolt.* **2021**, *11*, 806–818. [[CrossRef](#)]
10. Khaldi, B.; Harrou, F.; Cherif, F.; Sun, Y. Monitoring a robot swarm using a data-driven fault detection approach. *Robot. Auton. Syst.* **2017**, *97*, 193–203. [[CrossRef](#)]
11. Rostek, K.; Morytko, Ł.; Jankowska, A. Early detection and prediction of leaks in fluidized-bed boilers using artificial neural networks. *Energy* **2015**, *89*, 914–923. [[CrossRef](#)]
12. Zhang, S.; Shen, G.; An, L.; Gao, X. Power station boiler furnace water-cooling wall tube leak locating method based on acoustic theory. *Appl. Therm. Eng.* **2015**, *77*, 12–19. [[CrossRef](#)]
13. An, L.; Wang, P.; Sarti, A.; Antonacci, F.; Shi, J. Hyperbolic boiler tube leak location based on quaternary acoustic array. *Appl. Therm. Eng.* **2011**, *31*, 3428–3436. [[CrossRef](#)]
14. Zhang, J.; Fariborz, H. Development of Artificial Neural Network based heat convection algorithm for thermal simulation of large rectangular cross-sectional area Earth-to-Air Heat Exchangers. *Energy Build.* **2010**, *42*, 435–440. [[CrossRef](#)]
15. Ji, H.; He, X.; Shang, J.; Zhou, D. Incipient sensor fault diagnosis using moving window reconstruction-based contribution. *Ind. Eng. Chem. Res.* **2016**, *55*, 2746–2759. [[CrossRef](#)]
16. Song, B.; Tan, S.; Shi, H. Key principal components with recursive local outlier factor for multimode chemical process monitoring. *J. Process Control* **2016**, *47*, 136–149. [[CrossRef](#)]
17. Ge, Z.; Song, Z. Online monitoring of nonlinear multiple mode processes based on adaptive local model approach. *Control Eng. Pract.* **2008**, *16*, 1427–1437. [[CrossRef](#)]
18. Guo, J.; Wang, X.; Li, Y.; Wang, G. Fault detection based on weighted difference principal component analysis. *J. Chemom.* **2017**, *31*, e2926. [[CrossRef](#)]
19. Rashid, M.M.; Yu, J. Hidden Markov model based adaptive independent component analysis approach for complex chemical process monitoring and fault detection. *Ind. Eng. Chem. Res.* **2012**, *51*, 5506–5514. [[CrossRef](#)]
20. Zhao, S.J.; Zhang, J.; Xu, Y.M. Monitoring of processes with multiple operating modes through multiple principle component analysis models. *Ind. Eng. Chem. Res.* **2004**, *43*, 7025–7035. [[CrossRef](#)]

21. Zhu, J.; Ge, Z.; Song, Z. Robust supervised probabilistic principal component analysis model for soft sensing of key process variables. *Chem. Eng. Sci.* **2015**, *122*, 573–584. [[CrossRef](#)]
22. Beaver, S.; Palazoglu, A. A cluster aggregation scheme for ozone episode selection in the San Francisco, CA Bay Area. *Atmos. Environ.* **2006**, *40*, 713–725. [[CrossRef](#)]
23. Tong, C.; Palazoglu, A.; Yan, X. An adaptive multimode process monitoring strategy based on mode clustering and mode unfolding. *J. Process Control* **2013**, *23*, 1497–1507. [[CrossRef](#)]
24. Cai, D.; He, X.; Han, J. Document clustering using locality preserving indexing. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 1624–1637. [[CrossRef](#)]
25. Ge, Z.; Yang, C.; Song, Z.; Wang, H. Robust online monitoring for multimode processes based on nonlinear external analysis. *Ind. Eng. Chem. Res.* **2008**, *47*, 4775–4783. [[CrossRef](#)]
26. Zhu, Z.; Song, Z.; Palazoglu, A. Process pattern construction and multi-mode monitoring. *J. Process Control* **2012**, *22*, 247–262. [[CrossRef](#)]
27. Feng, J.; Li, K. MRS-kNN fault detection method for multirate sampling process based variable grouping threshold. *J. Process Control* **2020**, *85*, 149–158. [[CrossRef](#)]
28. Ng, Y.S.; Srinivasan, R. An adjoined multi-model approach for monitoring batch and transient operations. *Comput. Chem. Eng.* **2009**, *33*, 887–902. [[CrossRef](#)]
29. Lu, N.; Gao, F.; Wang, F. Sub-PCA modeling and on-line monitoring strategy for batch processes. *AIChE J.* **2004**, *50*, 255–259. [[CrossRef](#)]
30. Hwang, D.H.; Han, C. Real-time monitoring for a process with multiple operating modes. *Control Eng. Pract.* **1999**, *7*, 891–902. [[CrossRef](#)]
31. Zhao, C.; Wang, F.; Lu, N.; Jia, M. Stage-based soft-transition multiple PCA modeling and on-line monitoring strategy for batch processes. *J. Process Control* **2007**, *17*, 728–741. [[CrossRef](#)]
32. Yao, Y.; Gao, F. Phase and transition based batch process modeling and online monitoring. *J. Process Control* **2009**, *19*, 816–826. [[CrossRef](#)]
33. Ge, Z.; Song, Z. A distribution-free method for process monitoring. *Expert Syst. Appl.* **2011**, *38*, 9821–9829. [[CrossRef](#)]
34. Zhang, C.; Gao, X.; Xu, T.; Li, Y. Nearest neighbor difference rule-based kernel principal component analysis for fault detection in semiconductor manufacturing processes. *J. Chemom.* **2017**, *31*, e2888. [[CrossRef](#)]
35. Breunig, M.M.; Kriegel, H.P.; Ng, R.T.; Sander, J. LOF: Identifying density-based local outliers. In Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, Philadelphia, PA, USA, 16–18 June 2000; pp. 93–104. [[CrossRef](#)]
36. Lee, J.; Kang, B.; Shin, K.; Kang, S. Online process monitoring scheme for fault detection based on independent component analysis (ICA) and local outlier factor (LOF). In Proceedings of the 40th International Conference on Computers & Industrial Engineering, Awaji Island, Japan, 25–28 July 2010; pp. 1–6.
37. Alghushairy, O.; Alsini, R.; Soule, T.; Ma, X. A review of local outlier factor algorithms for outlier detection in big data streams. *Big Data Cogn. Comput.* **2020**, *5*, 1. [[CrossRef](#)]
38. Guo, J.; Wang, X.; Li, Y. kNN based on probability density for fault detection in multimodal processes. *J. Chemom.* **2018**, *32*, e3021. [[CrossRef](#)]
39. Ma, H.; Hu, Y.; Shi, H. Fault detection and identification based on the neighborhood standardized local outlier factor method. *Ind. Eng. Chem. Res.* **2013**, *52*, 2389–2402. [[CrossRef](#)]
40. Rosenblatt, M. Curve estimates. *Ann. Math. Stat.* **1971**, *42*, 1815–1842. [[CrossRef](#)]
41. Parzen, E. On estimation of a probability density function and mode. *Ann. Math. Stat.* **1962**, *33*, 1065–1076. [[CrossRef](#)]
42. Lee, J.; Kang, B.; Kang, S.H. Integrating independent component analysis and local outlier factor for plant-wide process monitoring. *J. Process Control* **2011**, *21*, 1011–1021. [[CrossRef](#)]
43. Hsu, C.C.; Chen, L.S.; Liu, C.H. A process monitoring scheme based on independent component analysis and adjusted outliers. *Int. J. Prod. Res.* **2010**, *48*, 1727–1743. [[CrossRef](#)]
44. Odiowei, P.E.P.; Cao, Y. Nonlinear dynamic process monitoring using canonical variate analysis and kernel density estimations. *IEEE Trans. Ind. Inform.* **2009**, *6*, 36–45. [[CrossRef](#)]
45. Chen, Q.; Wynne, R.J.; Goulding, P.; Sandoz, D. The application of principal component analysis and kernel density estimation to enhance process monitoring. *Control Eng. Pract.* **2000**, *8*, 531–543. [[CrossRef](#)]
46. Mori, J.; Yu, J. Quality relevant nonlinear batch process performance monitoring using a kernel based multiway non-Gaussian latent subspace projection approach. *J. Process Control* **2014**, *24*, 57–71. [[CrossRef](#)]
47. Basu, P.; Fraser, S.A. *Circulating Fluidized Bed Boilers: Design and Operation*, 1st ed.; Elsevier Science: Amsterdam, The Netherlands; Butterworth-Heinemann: Oxford, UK, 1991; ISBN 0-7506-9226-X.
48. Li, Y.S.; Sanchez-Pasten, M.; Spiegel, M. High temperature interaction of pure Cr with KCl. In *Materials Science Forum*; Trans Tech Publications, Ltd.: Schwyz, Switzerland, 2004; Volume 461, pp. 1047–1054.
49. Back, S.K.; Yoo, H.M.; Jang, H.N.; Joung, H.T.; Seo, Y.C. Effects of alkali metals and chlorine on corrosion of super heater tube in biomass circulating fluidized bed boiler. *Appl. Chem. Eng.* **2017**, *28*, 29–34.
50. Anthony, E.J.; Iribarne, A.P.; Iribarne, J.V. A new mechanism for FBC agglomeration and fouling when firing 100% petroleum coke. In Proceedings of the 13th International Conference on FBC, Orlando, FL, USA, 7–10 May 1995; pp. 523–533.

- 
51. Anthony, E.J.; Jia, L. Agglomeration and strength development of deposits in CFBC boilers firing high-sulfur fuels. *Fuel* **2000**, *79*, 1933–1942. [[CrossRef](#)]
  52. Lin, W.; Krusholm, G.; Dam-Johansen, K.; Musahl, E.; Bank, L. Agglomeration phenomena in fluidized bed combustion of straw. In Proceedings of the 14th International Conference on FBC, Vancouver, BC, Canada, 11–16 May 1997; pp. 831–838.