electronics

MDPI

Article

Fault Classification of Nonlinear Small Sample Data through Feature Sub-Space Neighbor Vote

Xian Du [1,*], Jingyang Yan [1] and Rui Ma [2]

[1] Department of Mechanical and Industrial Engineering, Institute for Applied Life Sciences, College of Information and Computer Sciences, University of Massachusetts, Amherst, MA 01003, USA; jingyangyan@umass.edu

[2] Department of Electrical and Computer Engineering, Institute for Applied Life Sciences, University of Massachusetts, Amherst, MA 01003, USA; ruima@umass.edu

* Correspondence: xiandu@umass.edu

Received: 26 October 2020; Accepted: 17 November 2020; Published: 19 November 2020

Abstract: The fault classification of a small sample of high dimension is challenging, especially for a nonlinear and non-Gaussian manufacturing process. In this paper, a similarity-based feature selection and sub-space neighbor vote method is proposed to solve this problem. To capture the dynamics, nonlinearity, and non-Gaussianity in the irregular time series data, high order spectral features, and fractal dimension features are extracted, selected, and stacked in a regular matrix. To address the problem of a small sample, all labeled fault data are used for similarity decisions for a specific fault type. The distances between the new data and all fault types are calculated in their feature subspaces. The new data are classified to the nearest fault type by majority probability voting of the distances. Meanwhile, the selected features, from respective measured variables, indicate the cause of the fault. The proposed method is evaluated on a publicly available benchmark of a real semiconductor etching dataset. It is demonstrated that by using the high order spectral features and fractal dimensionality features, the proposed method can achieve more than 84% fault recognition accuracy. The resulting feature subspace can be used to match any new fault data to the fingerprint feature subspace of each fault type, and hence can pinpoint the root cause of a fault in a manufacturing process.

Keywords: fault classification; small labeled sample; high dimension

1. Introduction

The emerging industry 4.0 is based on smart sensors that monitor a complex manufacturing process and machine intelligence techniques that automate the process control by extracting knowledge from the sensing data. A typical intelligent sensing application is the automation of fault diagnosis. The automation of fault diagnosis can effectively eliminate the laborious, tedious, and erroneous process of maintenance, and improve the efficiency of manufacturing process quality control. Generally, an intelligent fault detection technique can first use prior knowledge and sensing data to label faulty data from large amounts of normal data of the manufacturing process. Then, a fault classification method will classify the faulty data into specific types for further identification of fault causes and correction.

Many intelligent data-driven classification techniques have been proposed for fault detection of manufacturing processes given multidimensional time-series sensing data, including Principal Component Analysis (PCA) [1], *k*-NN [2–4], Support Vector Machine (SVM) [5,6], and the Gaussian Mixture Model (GMM) [7,8]. Most of these techniques are designed for Gaussian, linear and regular time-series sensing data. In practice, however, many sensing data are non-Gaussian, nonlinear, with unequal lengths and unsynchronized batch trajectories [1].

Du [2] demonstrated that high order spectral (HOS) features and further PCA extraction can significantly improve the fault detection accuracy and lower the false alarm rate of one-class classifiers for nonlinear and non-Gaussian sensing data. The proposed one-class *k*-NN classifier can achieve a 100% fault detection rate without causing any false alarm on the well-known semiconductor etching process data [1]. Further classification of the faulty data involves the follow-up work of the whole fault diagnosis process for the root cause analysis of the faults. In many practical manufacturing processes, fault data sizes are small or of a limited number because acquisition and labeling of faulty data samples are costly. The small size of labeled fault data hinders the training of a conventional pattern classifier for fault classification. Moreover, the increment of sensor numbers will cause "the curse of dimensionality", the high dimensionality of sensing data diminishing the relative contrast between near and far neighbors [9,10]. The concentration effect of the distance measure thus reduces the usability of distance metrics for discriminating faults from each other. In other words, the presence of irrelevant attributes conceals the contributing features and causes biases and errors in the classification of fault data. Addressing the presence of irrelevant attributes and taking into consideration subsets of attributes to define outliers is one of the main solutions for tackling challenges specific to high-dimensional data in Euclidean space.

To classify small and high dimensional fault data of nonlinearity and non-Gaussianity, the following innovative methods are proposed:

- To select the features that are most relevant to a specific fault type, a similarity-based sub-space neighborhood of the fault is created from the training data.
- To address a small fault sample size, all labeled data of a specific fault are used in its feature subspace, as the similarity neighborhood of the fault, to measure the closeness of new fault data to this fault. A similarity metric will measure the probability of the new data falling in a specific fault neighborhood. The fault data are labeled as a fault type that has the largest similarity probability. Meanwhile, the selected features of each fault type work as the fingerprint of this specific fault and can pinpoint the root cause of the fault.
- To capture the dynamics, nonlinearity, and non-Gaussianity in the irregular time series data for vector and matrix computation, the magnitude and entropy of the relative power over a frequency range and fractal dimension (FD) features are extracted and stacked to a regular data structure.

The high accuracy and robustness of the method are demonstrated with experimental results on a publicly available semiconductor process dataset.

In Section 2, the state-of-the-art fault classification techniques for a small sample of high dimensions are presented. In Section 3, a feature selection method based on HOS features and a fractal dimension is presented. In Section 4, a feature subspace-based fault classification algorithm is proposed. Section 5 describes the performance evaluation of the classification techniques for a semiconductor data set. In Section 6, the experiments and proposed methods are concluded and discussed.

2. The State of the Art

Data-driven fault classification approaches identify the fault types of new fault data by using various pattern classification models and feature extraction algorithms. These methods need little prior knowledge of faults for implementation. The popular data-driven fault classification methods include PCA, SVM, Fisher discriminant analysis (FDA), *k*-NN, and Artificial Neural Networks (ANNs).

Conventional PCA methods linearly transform high-dimensional and high-correlated data into space where uncorrelated variables have the most variations along their corresponding directions in the data [11]. These methods are designed for linear and Gaussian data classification; however, they have limited accuracy for fault detection and classification of high nonlinear and non-Gaussian manufacturing processes [1]. To address the nonlinear process data, kernel PCA (KPCA) methods have been proposed [12–15]. By applying a kernel function, the nonlinear raw data are mapped into a linear higher-dimensional feature space. Then, principal components in the feature space can be extracted

by a traditional PCA process. Lee et al. [12] proposed a multiway KPCA technique to avoid complex neural network optimization. Alcala and Qin [16] calculated reconstruction-based contributions (RBCs) to diagnose faults in nonlinear principal component models based on KPCA. Deng et al. [17] developed a two-step localized KPCA based incipient fault diagnosis for nonlinear industrial processes. The two-step KPCA preserves both the global and local data structure information. Lee et al. [18] proposed a tool wear/tool condition monitoring system by using KPCA and kernel density estimation. Though PCA has shown accurate applications for fault detection, it cannot produce good results for fault classification easily as the correlation between fault types is not considered in PCA transform.

Instead, FDA aims to optimally classify faults by minimizing the separation within classes and maximizing the separation between classes. The basis for the FDA fault classification is the conjecture that different types of faults are linear combinations of tool-state variables and the best linear combination of variables normally can be determined by the regression of the tool-state data. Conventional FDA is used as a linear classifier. Goodlin et al. [19] used an orthogonal linear discriminant approach (LDA) and fault-specific control charts to simultaneously detect and classify a fault. Verron et al. [20] improved the LDA classification performance by selecting important features based on mutual information between variables. Moreover, Fuente et al. [21] used LDA for detecting and diagnosing faults in a real plant by an optimal lower-dimensional representation of each fault type of data. Such a linear classification method is inappropriate for the fault recognition of a nonlinear manufacturing process data. Additionally, LDA cannot address the fault diagnosis of a small training dataset and high dimensional processes. To deal with nonlinear classifications, kernel functions have been introduced to map data from the original space to a high-dimensional feature space, which is based on a nonlinear mapping function, such as polynomial function or Gaussian radial basis function (RBF). Lu and Yan [22] proposed a method based on kernel FDA (KFDA) and self-organizing map networks (SOM) to improve the visualization of process monitoring. First, they applied KFDA analysis to map data into high-dimensional space and the optical Fisher feature vector is extracted to distinguish the normal state and different kinds of faults. Then, the Fisher feature vector space is visualized by applying SOM. Meanwhile, variant counterparts of KFDA have been proposed for fault classifications [23,24]. Ge et al. [24] proposed a kernel-based semi-supervised FDA model for nonlinear fault classification of a limited number of labeled data samples. Their KFDA model is constructed by including both labeled and unlabeled data samples in the semi-supervised data matrix and results in better classification performance than LDA models and the local FDA [23]. Adil et al. [25] proposed the application of exponential discriminant analysis (EDA) to overcome small sample size problems and improve the fault classification capability of discriminant analysis methods. The EDA approach used matrix exponential of within-class and between-class scatter matrices in the discriminant function. However, its discriminant function for each class has the same assumption as the LDA that the data for each class is normally distributed, which limits its classification accuracy when fault data are non-Gaussian.

Unlike the FDA, SVM methods have no assumption that the data are normally distributed. SVM methods are based on the structural risk minimization principle in the statistical learning theory. The greatest advantage of SVM for fault diagnosis is that it is suitable for a small sample-based decision, as it can maximally excavate the implicit classification knowledge in the data [26]. The initial idea of SVM is to use a linear separating hyperplane to divide the training samples into two classes. In the case that the samples are not linearly separable, kernel functions can be used to nonlinearly project the data samples into a higher dimensional feature space for linear classification. Cho and Jiang [27] designed a reproducing kernel to map the classified fault residuals into points in the reproducing kernel Hilbert space so that the residuals among different fault classes can be separated by the widest margin. The fault residuals are generated by comparing the measured outputs with the predicted ones based on an analytical model of the plant. Jan et al. [28] used SVM and ten time-domain statistical features to classify sensor faults. Their receiver operating characteristics (ROC) curve shows the efficiency of SVM over a neural network. Yang et al. [29] used RBF-kernel SVM (KSVM) classifiers and FD features for fault diagnosis of rolling element bearing. FD can quantitatively describe the non-linear

behavior of a vibration signal. In [30], permutation entropy (PE) of the vibration signal is calculated to detect the malfunctions of a bearing. SVM is then optimized by inter-cluster distance (ICD) in the feature space (ICDSVM) and used to recognize the fault type, as well as fault severity. Li et al. [31] proposed a deep stacking least squares SVM (LS-SVM) for rolling bearing fault diagnosis. As a variant of SVM, LS-SVM attempts to minimize the least squares errors and margin errors at the same time. The stacking-based representation learning (S-RL) was applied to train the classifier. S-RL can adaptively extract corresponding fault features from original data. In [32], Aoyagi et al. proposed a simple method to construct process maps for additive manufacturing using SVM. They also found that the value of a decision function in SVM had a physical meaning which may be a semi-quantitative guideline for porosity density of parts fabricated by additive manufacturing. Additionally, various counterparts of the KSVM models have recently been applied successfully to fault classifications in nonlinear complex nuclear reactors [33], motor systems [34], and so on.

As the simplest nonlinear data-driven methods for fault classification, *k*-NN, based on distances, classifies an unknown instance by correlating it with a known instance through a similarity function or an effective distance. The *k*-NN classifiers have shown high fault detection rates for nonlinear and non-Gaussian time series data [2–4]. Shi et al. [35] proposed a reinforced *k*-NN method in for chatter identification in high-speed milling. They applied the idea of reinforcement learning of punishing and rewarding into *k*-NN to keep the effectiveness of the chatter identification model. In [36], Sun et al. combined *k*-NN and PCA for fault detection in a semiconductor manufacturing process. The PCA algorithm was employed to reduce the data dimension and *k*-NN was applied to classify normal data and faulty data. Moreover, *k*-NN has been applied to fault classification in industrial processes [3], motors [37], and so on.

Additionally, Han et al. [38] comparatively evaluated SVM, ANN, and random forest applications in fault diagnoses, especially in rotating machinery. SVM and ANN were both shown to be less robust to features than the random forest, especially with a small training set. In the case of a small sample, the neural network algorithm often shows a poor generalization ability, namely the over-fitting problem [26].

For a small sample of high dimensions, many data-driven classification techniques cannot perform effectively because of insufficient training data and redundant feature space. The general solution is to cluster the data using a subspace of the features. For example, Cheng and Church [39] classify a small sample of high dimensions by identifying the subsets of data and the subspace of features with high similarity scores. Their proposed biclustering method allows the simultaneous clustering of data and features of a matrix based on variance. Such algorithms have been successfully applied to biological gene expression data [40]. Motivated by biclustering algorithms, a solution is proposed for fault feature selection and fault classification based on the similarity scores of the fault data and each fault feature subspace.

3. Feature Extraction of Nonlinear Time Series Data

Figure 1a shows three-dimensional multivariate time series data that can be acquired from many manufacturing processes. The data have an unequal batch and step length and unsynchronized batch trajectory. AHOS feature extraction can be applied to each variable of the data to integrate the variables and time into spectral features in one dimension [2]. The proposed HOS features include the Mean magnitude (Mag) of the spectrum and Entropy (Ent). These features contain the dynamic information of a process, such as the phase estimation of non-Gaussian parametric signals, and the nonlinear properties of mechanisms that generate time series via phase relations of their harmonic components. Hence, the HOS features have been successfully used for fault detection [2,41–44]. Additionally, the FD feature is extracted in this paper to enrich the characteristics of faults for separating faults between different fault types. FD can interpret observations of physical systems where the time trace of the measured quantities is irregular [29,45].

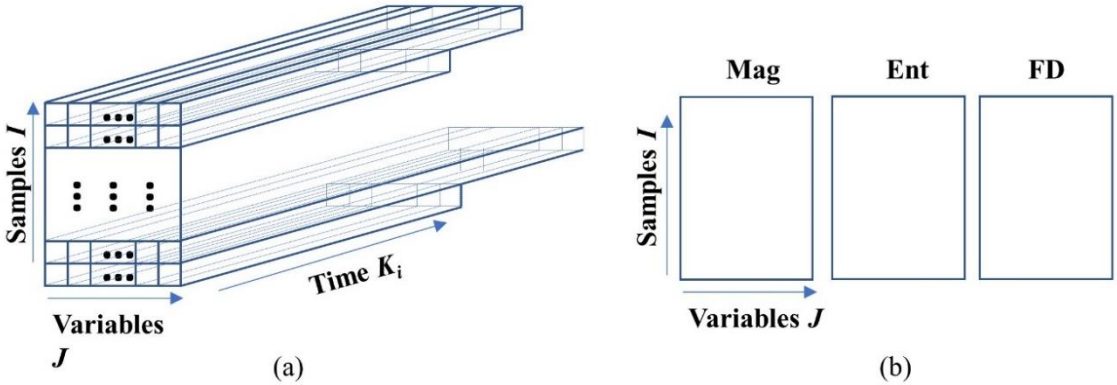


Figure 1. Feature extraction and batch-wise unfolding of time series. (**a**) Raw data with unequal sample duration; (**b**) unfolding result by high order spectral (HOS) bispectral features Mean magnitude (Mag) and Entropy (Ent), and fractal dimension (FD).

As shown in Figure 1b, each set of the features extracted from all variables will be put side by side in the sequence of Mag, Ent, and FD. An unfolding result in the format of a two-dimensional matrix will be formed with the size of $I \times (J \times H + J)$. Here, H is the number of spectral features, and $H = 2$ in this paper.

3.1. HOS Features

The HOS feature generally refers to the third and fourth-order spectrum. These features can measure the deviations of series-time data deviations of linearity, stationarity, and Gaussianity. Bispectrum, one of the popular HOS features, can quantify the skewness of a signal, which makes it popular in characterizing nonlinear data. Bispectrum can be calculated by the two-dimensional Fourier transform of third order cumulants as follows,

$$B(f_1, f_2) = E[F(f_1) \cdot F(f_2) \cdot F^*(f_1 + f_2)] \tag{1}$$

Here, f_1 and f_2 are two independent frequencies, $F(f)$ is the Fourier transform of the random signal X, $E[.]$, and '*', respectively, are the expectation operation and the complex conjugate. The frequencies are normalized to a 0–1 range by the Nyquist frequency. Further, the amplitude of the bispectrum at the frequency (f_1, f_2) measures the coupling effects between the spectral components at the frequencies f_1, f_2, and $f_1 + f_2$. The mean of the magnitude of the bispectrum points within the region (f_1, f_2) can be calculated to quantify the non-linearity in the signal by,

$$\text{Mean\_mag} = \frac{1}{L_b}\left(\sum \left|B(f_1, f_2)\right|\right). \tag{2}$$

Here L_b is the number of bispectrum points within the region (f_1, f_2).

The large amplitude of the bispectrum indicates that a quadratic non-linearity exists in the signal. Moreover, bispectral entropies can be derived to characterize the rhythmic nature of time-series data such as the variability and irregularity of the signals. The normalized bispectral entropy (BE) is calculated by,

$$Ent = -\sum_n p_n \log p_n \tag{3}$$

Here, $p_n = \left(\left|B(f_1, f_2)\right|\right) / \left(\sum_\Omega \left|B(f_1, f_2)\right|\right)$. Ω refers to the non-redundant region between f_1 and f_2 in the bisepctrum map. The entropy is normalized to the 0–1 range (required of a probability). The entropy measures the uniformity of a power spectral distribution. A signal with a single frequency component has the smallest entropy, while a signal with all frequency components of equal power value (white noise) has the greatest entropy. The power spectrum of a measured signal is dominated

by a few peaks with low values in normal variations. The entropy of the power spectrum depends mainly on the degree of dominance of a few peaks, the number of the peaks, and their peakednesses. A stronger entropy value indicates a greater degree of signal irregularity.

3.2. FD Feature

Developed on the non-linear dynamics and chaos theory, FD is a ratio for statistically quantifying the complexity or irregularity of time series data. FD can be used to comparatively evaluate how a fractal pattern changes with the scale at which it is measured. The phase-plane and Poincare maps of chaotic systems have a fractal structure. The FD of the structure is a meaningful feature for recognition, classification, and diagnosis of such maps of chaos. For fault diagnosis, FD is a promising tool to interpret observations of physical systems where the time series of the measured quantities are irregular. The FD of the phase space trajectory of a dynamical system has several different types and various formal mathematical definitions. One commonly used FD method was proposed by Higuchi [46–48], which is defined by,

$$D = \frac{\log(L(K))}{\log(K)} \tag{4}$$

Here, for a time series $\{X(1), X(2), X(3), \ldots, X(Q)\}$, the segment length $L(K)$ is the length of the curve for interval time K. $L(K)$ is measured by the average value over K sets of $L_k(K)$. Here $L_k(K)$ is the length of each curve of a new time series that is reconstructed from the original time series $\{X(1), X(2), X(3), \ldots, X(Q)\}$. The new self-similar time series X_k^K and $L_k(K)$ are, respectively, calculated by,

$$X_k^K: X(k), X(k+K), X(k+2K), \ldots, X\left(k + \left[\frac{Q-k}{K}\right]K\right), k = 1, 2, \ldots, K \tag{5}$$

and,

$$L_k(K) = \left\{\left(\sum_{i=1}^{[(Q-k)/K]} \left|X(k+iK) - X(k+(i-1)K)\right|\right)\frac{Q-1}{[(Q-k)/K]K}\right\}/K \tag{6}$$

where k is an integer for the initial time, $[\cdot]$ is the integer part of the real number calculated by the operation in the square bracket.

4. Fault Classification by Nonlinear Feature Selection and Feature Sub-Space Neighbor Vote

Nearest neighbor is a simple and efficient nonlinear decision rule and often yields competitive results when compared with the other state-of-the-art classification methods. As a typical nearest neighbor method, the k-NN classifier labels a sample by calculating its distance to the normal training data. Given a confidence level-based distance threshold, the new sample data can be labeled as a fault if its distance to the k nearest neighbors of normal training data is greater than the threshold. The k-NN classifier has demonstrated a high fault detection rate and low false alarm rate for nonlinear and non-Gaussian time series data [2–4]. However, when the sample size of each fault type is small, especially imbalanced, it is difficult to obtain an appropriate number of k to apply k-NN for fault classification. For instance, k = 3 in [2] is not applicable for classifying fault data if the number of labeled training data in a fault type is less than four.

For a small labeled training sample, we propose to use the distance of new data to all labeled training data of any fault type for a similarity measure. To normalize the similarity between the new data and a fault type, the dissimilarity between the new data and all the other fault types can be integrated into the similarity metric as an inversely proportional factor.

The proposed fault classification algorithm hypothesizes that each fault type has an associated specific feature subspace that accounts for more similarity among the data within the fault type than the similarity between the data of this fault type and the other fault types. This feature subspace is composed of a set of specific data dimensions. Given the training data of any specific fault type, such a

feature subspace can be extracted by minimizing the neighborhood distance within the fault type while maximizing the neighborhood distance between it and the other fault types. Using the feature subspaces, the similarity between new query fault data and the labeled fault data of each fault type can be calculated. The maximization of the similarities between the query data and fault types leads to the most similar fault type of the data. Briefly, the proposed algorithm consists of two steps: extraction of the feature subspace of each fault type and classification of new query fault data.

4.1. Similarity-Based Feature Selection

To find the feature subspace of each fault type, we propose to search for features that have more similarities within the same fault than across fault types. The following ratio is defined to measure the similarity of the d_{th} feature and n_{th} fault type by,

$$Ratio(d;n) = \frac{\left(\sum\limits_{l=1,\ldots L}\left(\sum\limits_{l'=1,\ldots L,l'\neq l}\Delta x_d^{(l,l';n)}\right)\right)/(L(L-1))}{\sum\limits_{p=1,\ldots P}\left(\sum\limits_{l=1,\ldots L}\left(\sum\limits_{m=1,..,M}\Delta x_d^{(l,m;n,p)}\right)\right)/(LMP)} \tag{7}$$

Here, $\Delta x_d^{(l,l';n)}$ is the similarity between the l_{th} and the l'_{th} training data of the d_{th} feature within the n_{th} fault type, and $\Delta x_d^{(l,m;n,p)}$ is the similarity between the l_{th} training data of the n_{th} fault type and the m_{th} training data of the p_{th} fault type for the d_{th} feature. $l = 1, \ldots, L$; $l' = 1, \ldots, L$, $l' \neq l$; $m = 1, \ldots, M$; $n = 1, \ldots, N$; $p = 1, \ldots, P$ and $p \neq n$.

The $\Delta x_d^{(l,m;n,p)}$ is calculated by,

$$\Delta x_d^{(l,m;n,p)} = \left|x_d^{l;n} - x_d^{m;p}\right| \tag{8}$$

Here, $x_d^{l;n}$ is the value of the d_{th} feature of the l_{th} training data of the n_{th} fault type; $x_d^{m;p}$ is the value of the d_{th} feature of the m_{th} training data of the p_{th} fault type.

The $\Delta x_d^{(l,l';n)}$ is calculated by,

$$\Delta x_d^{(l,l';n)} = \left|x_d^{l;n} - x_d^{l';n}\right| \tag{9}$$

Here, $x_d^{l;n}$ and $x_d^{l';n}$ are, respectively, the values of the d_{th} feature of the l_{th} and l'_{th} training data of the n_{th} fault type.

If $Ratio(d;n) < 1$, the dimension d is counted in the feature subspace of the n_{th} fault type.

4.2. Feature Sub-Space Neighbor Vote for Fault Classification

The feature selection method results in a feature subspace D_i for the i_{th} fault type. Given any query fault data X^{query}, its similarity to the i_{th} set of classified fault data can calculated by,

$$\Delta x_{D_i}^{(query;i)} = \frac{1}{M_i}\sum_{m_i}\left|X_{D_i}^{query} - X_{D_i}^{m_i}\right| \tag{10}$$

where m_i is the index of data in the i_{th} fault type, M_i is the number of fault data of fault type i, X_i^{query} is the query data vector in the feature subspace of fault type i, and $X_i^{m_i}$ the m_i^{th} classified data vector in the data set of fault type i measured in the feature subspace of fault type i.

The similarity between the query data and the classified dataset of the j_{th} fault type in D_i, $j \neq i$, can be calculated by,

$$\Delta x_{D_i}^{(query;j)} = \frac{1}{M_j}\sum_{m_j}\left|X_{D_i}^{query} - X_{D_i}^{m_j}\right| \tag{11}$$

Here, m_j is the index of data in the j_{th} fault type, M_j is the number of data of fault type j, $X_{D_i}^{m_j}$ is the m_j^{th} classified data in the training data set of fault type j, $j \neq i$, measured in the feature subspace of fault type i.

Assuming the training data totally have I classes of faults, the quantitative comparison between the similarity of the query data and the fault type i and the similarity of the query data and all the other fault types can be measured by,

$$Ratio(i) = \frac{\Delta x_{D_i}^{(query;i)}}{\frac{1}{I-1}\sum_{j\neq i}\Delta x_{D_i}^{(query;j)}} \tag{12}$$

Here, I is the number of fault types. Similarly, we can calculate the $Ratio(k)$ for any other fault type, $k = 1, 2, \ldots, K$, $k \neq i$, in its feature subspace,

$$Ratio(k) = \frac{\Delta x_{D_k}^{(query;k)}}{\frac{1}{I-1}\sum_{j\neq k}\Delta x_{D_k}^{(query;j)}} \tag{13}$$

If the query data are fault type i, the query data should be closer to the data of the i_{th} fault type in D_i than to any other fault type in D_i, according to the feature subspace calculation in Section 4.1. To normalize the calculation of the closeness distance, the denominators in Equations (12) and (13) are introduced. Both the denominators are calculating the distances between the query data and the fault types not in the subspaces of these fault types. Hence the denominators will bring little bias into the ratios.

Considering the data noises, the probability that the query data belong to fault type i rather than type k can be estimated by,

$$Prob(i) = \frac{\#(Ratio(i) \leq Ratio(k))}{I-1} \tag{14}$$

Then, the fault type of the query data can be identified by seeking the maximum probability among all fault types by,

$$Fault\ type = \underset{i}{argmax}(prob(i)) \tag{15}$$

4.3. The Workflow of the Proposed Fault Classification Method

The query fault data that are identified can be stored in the data set of the fault type for new fault classification. The algorithm of fault classification is illustrated in Algorithm 1. Calibration data Xn are normal process data; fault data Xe are labeled for known fault types. The mean and standard deviation of the calibration data are the characteristics of the normal process, which are used to scale the fault data.

Algorithm 1 subspace fault classification

Input:
A new query data, calibration data X_n, fault data X_e,
Step 1: Find feature subspace for each fault
type:
Initialize:
Normalize the calibration data and fault data using the mean and standard deviation of the calibration data
For each fault type
For each feature
1. Calculate the similarity between the fault training data of this feature and the other types of fault training data of this feature by Equation (8).
2. Calculate the similarity within each type of fault training data of this feature by Equation (9).
3. Calculate the normalized cut $ratio(d;n)$ by Equation (7).
4. If $ratio(d;n) < 1$, count this feature in the feature subspace of the fault type
Step 2: Fault classification:
For the feature subspace corresponding to the i_{th} fault type, $i = 1, 2, \ldots, K$
1. Calculate the similarities between query data and the fault data types classified data set in their feature subspaces by Equations (10) and (11).
2. Calculate the similarity ratios by Equations (12)–(14).
3. Decide the fault type by Equation (15)
Step 3: Update the feature subspaces of fault type using Step 1.
Output: the fault type of a new query data, and its corresponding feature subspace.

5. Case Studies

In this section, the proposed fault classification method is evaluated using a simulation example and real manufacturing data from a semiconductor metal etch process [1]. Figure 2 illustrates the workflow of the evaluation process for the proposed feature selection and fault classification method. It mainly includes two steps: data preprocessing and fault classification. In data preprocessing, the missing data are cleaned in fault data. Then, HOS and FD features are extracted using Equations (1)–(6), and are stacked into a feature matrix, as shown in Figure 1b. The feature matrix is normalized to have mean 0 and standard deviation 1 using the mean and standard deviation of calibration data. In fault classification, one fault data used for testing and the remaining data for the Equations (1)–(6) to extract the fault feature subspace. The leave-one fault is used to test the classification accuracy by the Equations (7)–(15). For the whole dataset of each fault type, such a leave-one-out test will be calculated. The average accuracy of fault classification for each fault type is calculated.

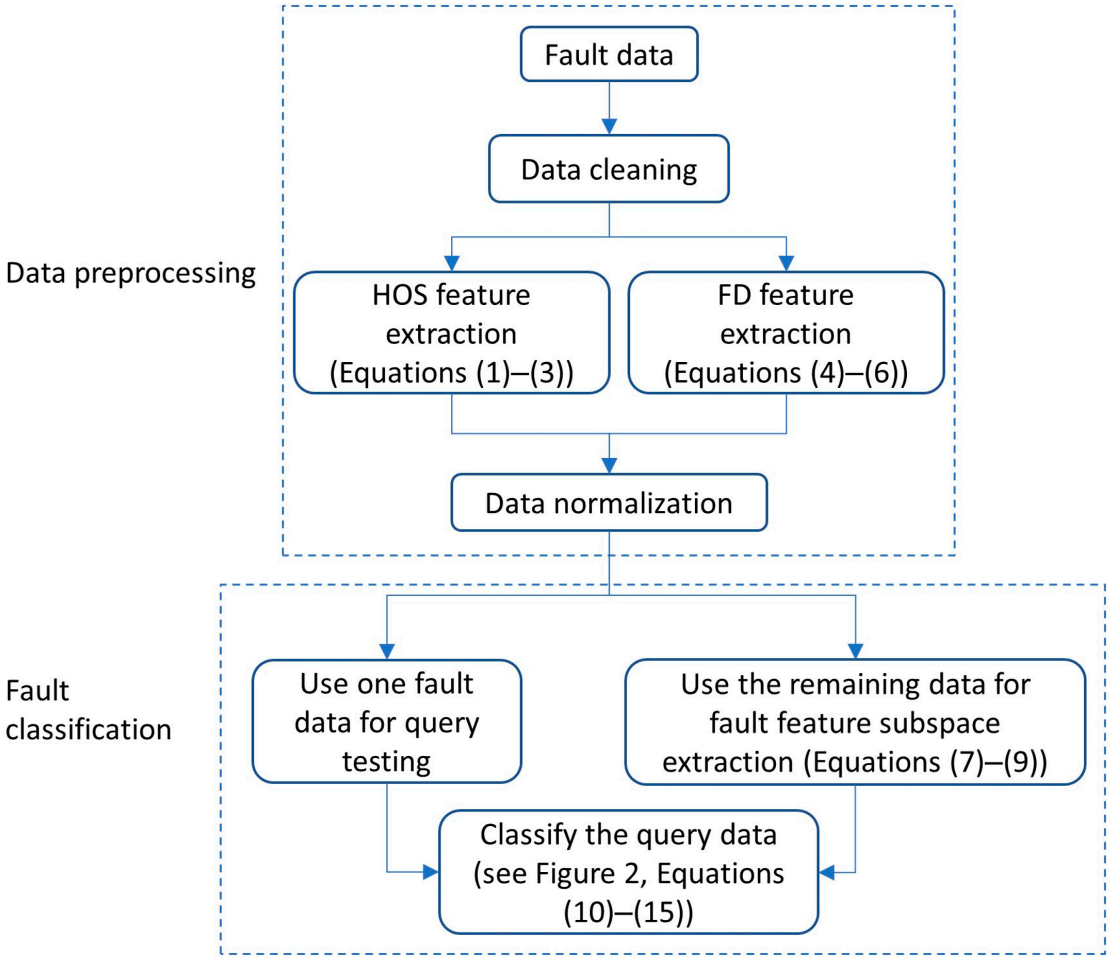


Figure 2. Workflow of the evaluation of the proposed fault isolation and identification method.

5.1. Simulation Example of a Two-Phase Batch Process

5.1.1. Data Set

A simulation example is generated to simulate a three-variable two-phase batch process [49]. The normal process has Phase 1,

$$\begin{aligned} x_1(t) &= x(t) + v_1 \\ x_2(t) &= x^2(t) - 3x(t) + v_2 \\ x_3(t) &= -x^3(t) + 3x^2(t) + v_3 \end{aligned} \tag{16}$$

Phase 2,

$$\begin{aligned} x_1(t) &= x(t) + v_1 \\ x_2(t) &= -x^3(t) - 3x(t) + v_2 \\ x_3(t) &= x^2(t) + 3x^2(t) + v_3 \end{aligned} \tag{17}$$

where k is the time index, x_1, x_2 and x_3 three variables measured in the batch process, t the driving variable uniformly distributed between (0.01, 2) in phase one and between (1.5, 4) in phase two, v_1, v_2, and v_3 random noise variables with Gaussian distributions with 0 mean and 0.01 variance. 100 samples are generated for each patch in three dimensions with 50–50 samples in each phase. To simulate two faulty batches, six ramp changes and six step changes are, respectively, introduced to the second and first variables in Equations (16) and (17) as follows,

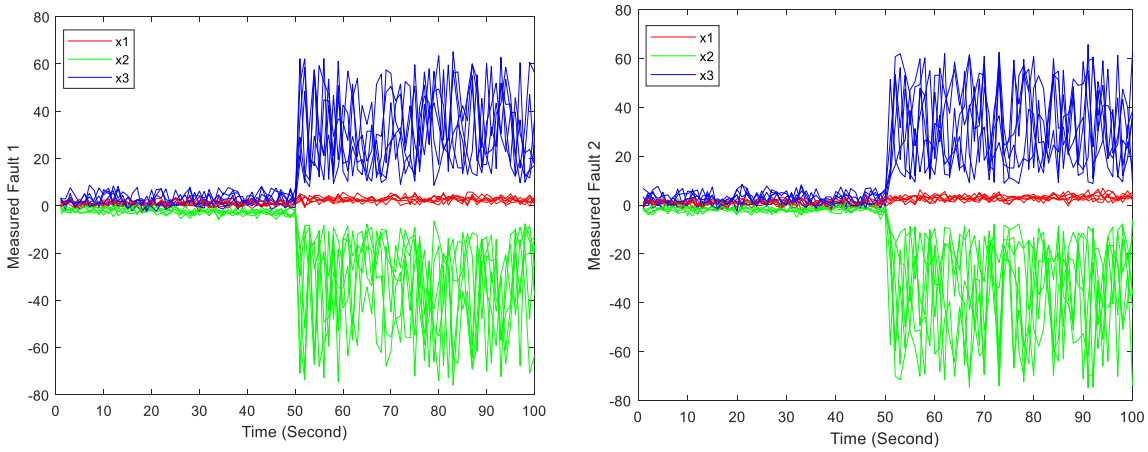
Phase one of fault one,

$$x_2(t) = x^2(t) - 3x(t) - 0.5\,d + v_2 \quad t \in [26, 50],\ d = 1, 2, 3, 4, 5, 6 \tag{18}$$

Phase two of fault two,

$$x_1(t) = x(t) + 0.025d(t - 75) + v_1 \quad t \in [76, 100],\ d = 1, 2, 3, 4, 5, 6 \tag{19}$$

Hence, a total of 12 patches are generated through Equations (18) and (19) for two fault types: six patches for every fault type. Figure 3 shows the data characteristics of different patches in two phases. The fault one (see Figure 3a) and fault 2 (see Figure 3b) cannot be apparently discriminated. As shown in [49], this two-phase batch process is non-Gaussian and nonlinear.

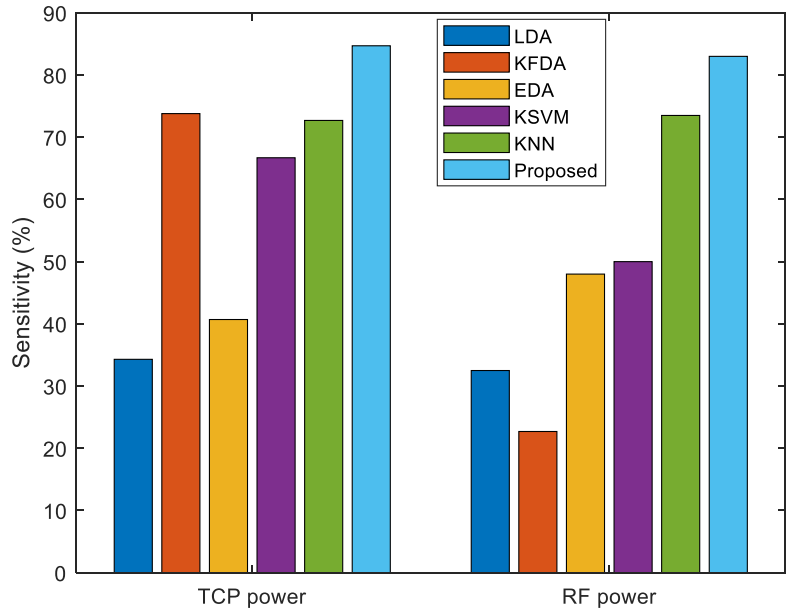


(**a**) (**b**)

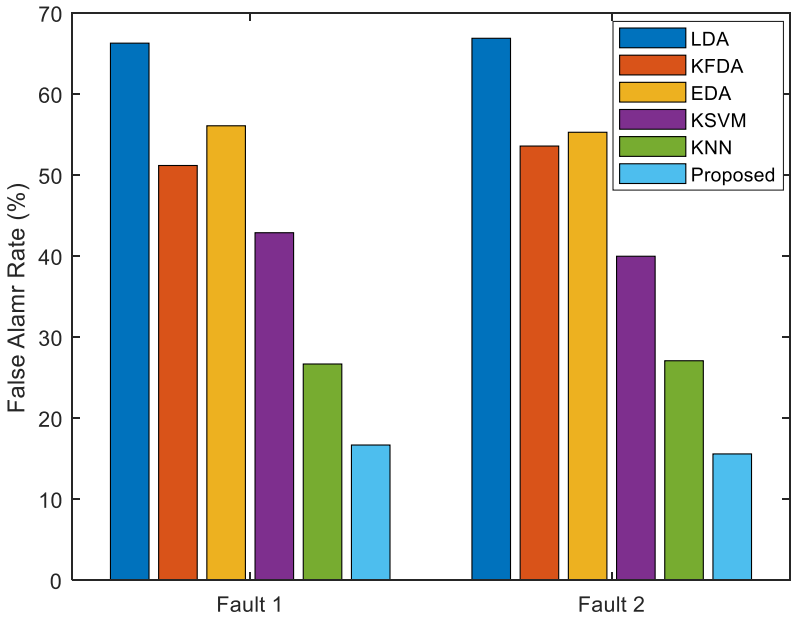
Figure 3. Simulation fault data in the two-phase batch process. (**a**) fault 1. (**b**) fault 2.

5.1.2. Fault Classification Results

Considering the randomness of the data generation, 100 samples are generated by the Equations (7)–(9). The simulation sample of each fault type form a 100 × 3 matrix. The HOS and FD features are extracted from every sample matrix and form a 100 × 9 matrix. Hence, the training and testing dataset of either fault type includes six 100 × 9 matrices. When applying the leave-one-out test workflow to the 12-fault data (see Figure 2), one fault is taken out from the whole fault datasets as a new query data. The remaining classified fault data are used for searching the feature subspace of each type of fault dataset by the Equations (7)–(9). For the new query data, the similarity between these data and each fault dataset in the feature subspace is calculated using the Equations (10)–(15). The fault classification results are given in Figure 4. The proposed method shows 84.7% and 83.0% fault classification rates that are significantly higher than any other methods, including LDA and four nonlinear fault classification methods including KFDA, EDA, KSVM, and k-NN (see Figure 4a) while maintaining the lowest false alarm rates of 16.7% and 15.6% for fault one and fault two (Figure 4b).



(**a**)



(**b**)

Figure 4. Fault classification results: (**a**) Sensitivity. (**b**) False alarm rate. LDA: linear discriminant approach; KFDA: kernel FDA; EDA: exponential discriminant analysis; KSVM: radial basis function (RBF)-kernel Support Vector Machine (SVM).

5.2. Semiconductor Metal Etch Process

5.2.1. Data Set

The proposed method is evaluated using the manufacturing data acquired from a semiconductor Al stack etch process performed on a Lam 9600 plasma etch tool. The goal of the process is to etch the TiN/Al-0.5% Cu/TiN/oxide stack with an inductively coupled BCl_3/Cl_2 plasma. The key controlled outputs of the process are the linewidth of the etched Al line, uniformity across the wafer, and the oxide loss. As wafer state sensors were not available in the original equipment, process state sensors

were selected and built into the processing tool to infer wafer state information [1]. These sensors mainly measured the engineering variables such as gas flow rates, chamber pressure, and RF power. The non-setpoint process variables with some normal variation were used for machine state monitoring, as shown in Table 1. These 19 variables were selected because of their highest relevance to the process and final product state based on the physics of the problem. The data were collected and recorded at 1 s intervals during etching for every sensor.

Table 1. Machine state variables used for monitoring semiconductor metal etch process.

1	BCl_3 Flow	2	Cl_2 Flow	3	RF Btm Pwr	4	RF Btm Rfl Pwr
5	Endpt Al	6	He press	7	Pressure	8	RF tuner
9	RF load	10	RF phase err	11	RF Pwr	12	RF impedance
13	TCP tuner	14	TCP phase err	15	TCP impedance	16	TCP top Pwr
17	TCP Rfl Pwr	18	TCP load	19	Vat valve	-	-

"Btm", "bottom"; "Pwr", "power"; "Rfl", "reflected"; "Endpt", "end-point".

Faults were intentionally induced in a series of three experiments by changing TCP power, RF power pressure, BCl3/Cl2 flow rate, and He chuck pressure. The final machine state data have a total of 129 wafers and 21 faults. To mimic the actual sensor failure, values for the controlled variables were intentionally moved off its setpoints and reset to normal. The resulting data perform like a controller seeing a biased sensor for a variable and adjusting accordingly. Hence, the variable value would appear normal, but it would not be. The effect of the sensor offset, however, will be evident in the remaining process variables because of the apparent physical relationship between the sensing variable and the other sensing variables. As shown in Table 2, six fault types of data were collected in the experiments. In each fault type, various fault magnitudes of sensor offset were induced to the fault sensors.

Table 2. Induced Fault Data.

Fault Sensor	Magnitude of Fault	# Faults
TCP power (W)	+50, +30, +20, +10, −10, −15	6
RF power (W)	+12, +10, +8, −12	4
Pressure (mTorr)	+3, +2, +1, −2	4
Cl_2 (sccm)	+5, −5, −10	3
BCl_3 (sccm)	+10, +5, −5	3
He chuck pressure	Not known	1

5.2.2. Data Preprocessing

The helium chuck fault and an RF power fault (faulty wafer 12, RF power +8) are removed from the fault data set because the former has an unknown magnitude and the latter has a large amount of missing data. Hence, only 19 faulty batches are used for fault classification in the present work.

Two HOS and one FD feature(s) are extracted from each variable of the raw data, which makes 57 features for each datum. Stacking all these features along the feature dimension, the transformed normal and fault data are, respectively, reconstructed into a 107×57 and a 19×57 matrix. From the normal data matrix, the mean and standard deviation of each of the 57 features for the normal process can be determined for normalizing fault data. Such a normalization of the fault data eliminates the effects of the process shift and standardizes the process data of each feature to one. Readers can refer to [2] for details of data preprocessing of normal raw data.

5.2.3. Fault Classification Results

During each fault classification procedure, one fault is taken out from the whole fault datasets as a new query data. The remaining classified fault data are used for searching the feature subspace of each type of fault dataset by the Equations (7)–(9). For the new query data, the similarity between these data

and each fault dataset in its feature subspace is calculated using the Equations (10)–(15). Applying the leave-one-out test workflow to the 19-fault data (see Figure 2), the probability matrix is obtained for each fault type as shown in Tables 3–7. Each column of Tables refers to the probability of a fault data identified by every feature subspace. For instance, in Table 3, the six TCP power fault data all have been classified as the TCP fault type with 100% probabilities. However, the RF power fault has two RF power datasets "+12" and "+10" being classified with 100% probabilities, while in the RF power data "−12" is mistakenly classified as "Cl_2" with 100% probability. Using Equation (15), the fault type is assigned to the fault data (feature subspace in Tables 3–7), which has the largest probabilities in each column. The highest probabilities in all the columns are highlighted in Tables 3–7. Overall, TCP power, pressure, and BCl_3 fault data are 100% accurately identified. The fault datasets of RF power and Cl_2, respectively, misclassify one and two fault datasets and therefore have 2/3 (66.6%) and 1/3 (33.3%) accuracy. The overall fault classification rate is 16/19 (84.2%). The three misclassified fault datasets create one false alarm for each fault type of Cl_2, RF power, and pressure (see Tables 4 and 6).

Table 3. Probability values obtained by Equation (13) for faulty TCP power dataset.

Feature Subspace	TCP Power (W) Fault					
	+50	**+30**	**+20**	**+10**	**−10**	**−15**
TCP power	1	1	1	1	1	1
RF power	0.5	0.75	0.5	0.5	0.5	0.5
Pressure	0	0	0	0	0	0
Cl_2	0.25	0.25	0.25	0.25	0.25	0.25
BCl_3	0.75	0.5	0.75	0.75	0.75	0.75

Table 4. Probability values obtained by Equation (13) for faulty RF power dataset.

Feature Subspace	RF Power (W) Fault		
	+12	**+10**	**−12**
TCP power	0.4	0.8	0.4
RF power	1	1	0.75
Pressure	0.25	0.25	0
Cl_2	0.75	0.5	1
BCl_3	0	0	0.25

Table 5. Probability values obtained by Equation (13) for faulty pressure dataset.

Feature Subspace	Pressure (mTorr) Fault			
	+3	**+2**	**+1**	**−2**
TCP power	0	0	0	0
RF power	0.5	0.5	0.25	0.5
Pressure	1	1	1	1
Cl_2	0.25	0.75	0.5	0.25
BCl_3	0.25	0.25	0.5	0.25

Table 6. Probability values obtained by Equation (13) for faulty Cl_2 dataset.

Feature Subspace	Cl_2 (sccm) Fault		
	+5	**−5**	**−10**
TCP power	0.5	0.5	0.5
RF power	0	0.25	1
Pressure	1	0.75	0.25
Cl_2	0.5	1	0.75
BCl_3	0.25	0.5	0

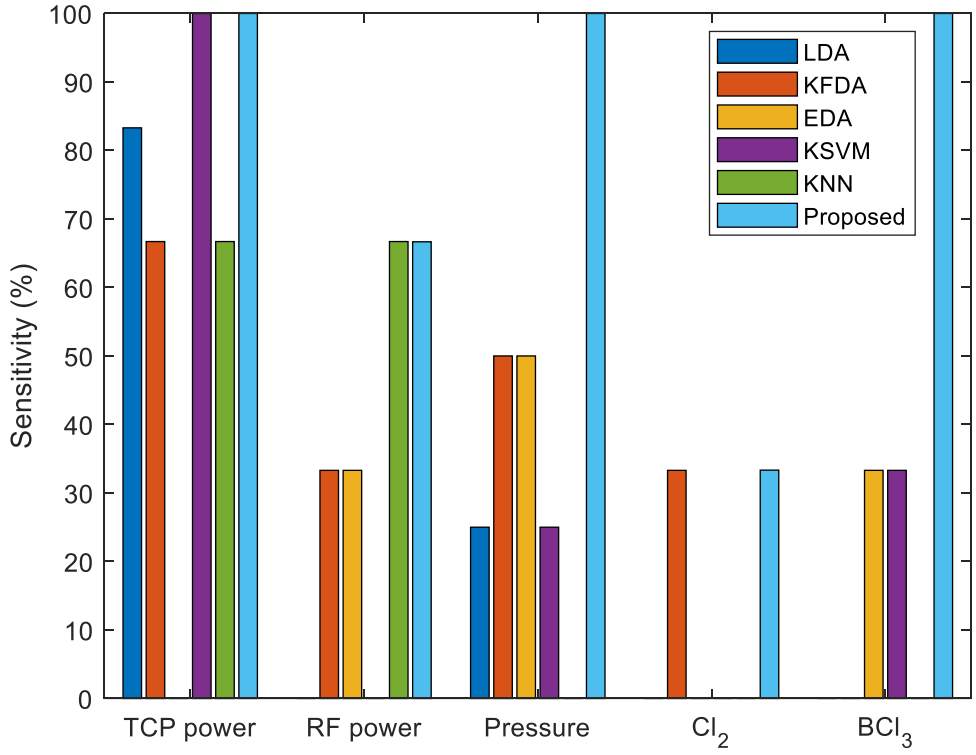
Table 7. Probability values obtained by Equation (13) for faulty BCl_3 dataset.

Feature Subspace	BCl_3 (sccm) Fault		
	+10	**+5**	**−5**
TCP power	0.75	0.75	0.75
RF power	0.5	0.75	0.25
Pressure	0.75	0	0
Cl_2	0.25	0.5	0.5
BCl_3	1	1	1

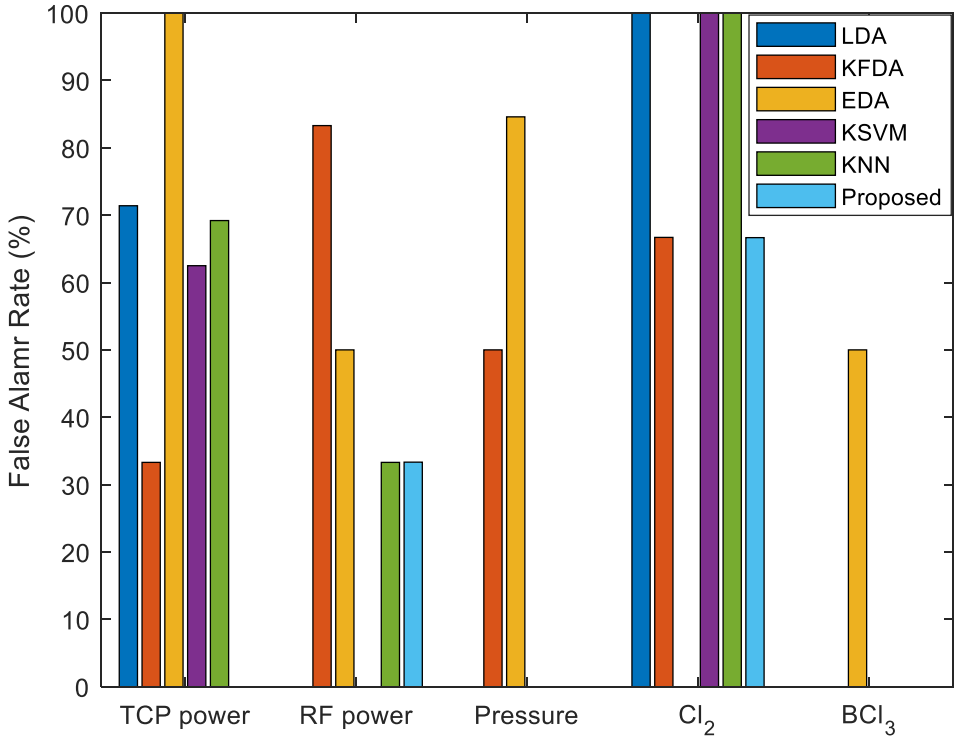
To better evaluate the performance of the proposed approach, the proposed method is compared with LDA and four nonlinear fault diagnosis methods including KFDA, EDA, KSVM, and *k*-NN using the 57 features. The leave-one-out cross-validation is applied on the 19-fault data and their labels. In other words, one fault datum is selected randomly for testing while the remaining data are used for training of the classification model.

The LDA and EDA classification methods assume that different classes generate data based on different Gaussian distributions. Hence, the parameters of a Gaussian distribution for each fault class are estimated in the training process. Then, the trained LDA and EDA classifiers find the fault type of a given fault dataset by searching for the fault class with the minimum expected classification cost. For a fair comparison of fault classification results, in KFDA and KSVM implementation, the polynomial kernel function of degree three is selected. The FDA and KSVM algorithms are applied to this high dimensional feature space to, respectively, obtain KFDA vectors and KSVM support vectors. For the *k*-NN classification, a given fault dataset is labeled by calculating its similarity with the various types of fault data in the training dataset. In this paper, some fault types only have two training datasets. Hence, $k = 2$.

Figure 5 is the comparative evaluation of different fault classification methods on the five types of fault data. As shown in Figure 5a, the proposed method has significantly higher sensitivities to isolate all five fault data types than the other classification methods. Especially, the proposed method using the 57 features has 100% sensitivity in classifying fault types of TCP power, pressure, and BCl_3. As shown in Figure 5b, the proposed method has false alarm rates of 0, 33.3%, 0, 66.7%, and 0 to classify the TCP power, RF power, pressure, Cl_2, and BCl_3. Note, the other classifiers demonstrate 0 false alarm rates for some fault types because they misrecognize the fault data of these fault types.



(**a**)



(**b**)

Figure 5. Comparison of different fault classification methods on the five types of fault data: (**a**) Sensitivity; (**b**) False alarm rate.

Although KSVM classification can also have 100% sensitivity for TCP power, it has zero classification accuracies for the RF power and Cl_2 fault classes. Such a biased fault classification result is caused by the significantly greater amount of TCP power fault data than the other four faults. Such a similar poor classification result also occurs when *k*-NN is implemented on the data, due to the smaller sample size causing more imprecise neighborhood description for the fault types.

The LDA, EDA and KFDA classification results have no zero accuracies for any fault types, while all sensitivity values are equal to or lower than 50%. Such a bad performance is mainly due to the assumption of Gaussian distribution in the LDA, EDA, and KFDA models that miscalculates the complex non-Gaussian physics in the semiconductor data.

Additionally, the proposed similarity classification method using only 38 HOS features also shows a significantly higher accuracy than the *k*-NN and LDA classification results, while it misses one TCP power fault and one RF power fault compared to the same classification method using both the 38 HOS features and 19 FD features.

5.2.4. Fault Feature Subspace

In this paper, the feature space consists of 19 Mean magnitudes of the spectrum (Mag), 19 Entropy (Ent), and 19 FD features. Each variable of the fault data has a Mag feature, Ent feature, and FD feature. The significance of these features of the variables composes a fingerprint for each fault type. This fingerprint is the key to the fault identification. As shown in Figure 6, the correlation is marked between variables, extracted features, and fault types as a fingerprint of each fault type for fault identification. Note that all 19-fault data are used to generate Figure 6 for the analysis of fault features. The denser color in the mark indicates that there are more contributions from the corresponding variable for the corresponding fault type. For example, the variable "He press" has all three features marked as the densest color for the fault types "TCP power", "RF power", "pressure", and "Cl_2", but only one feature is significant for the fault type "Cl_3". These marks explicitly tell which variables are significantly affected by a specific fault type. Such information is critical for fault diagnosis. The densities of the mark color indicate the weight of significance of the variable on the fault type. The proposed feature subspace classification method can be interpreted as a matching of any new fault data to the fingerprint of each fault type, and hence can pinpoint the root cause of a fault—which sensor was reset or faulty in the etch process in this paper.

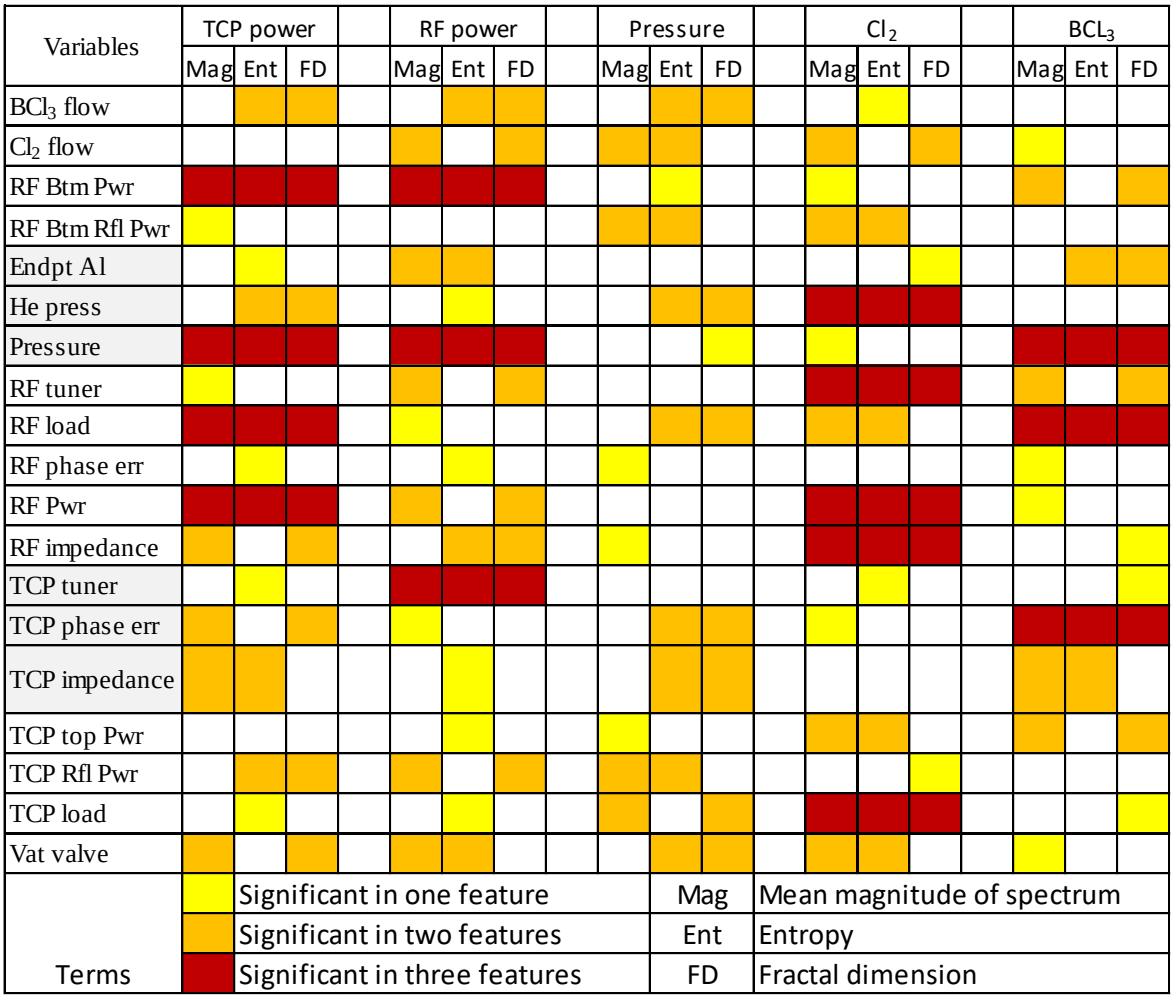


Figure 6. The diagnosis of process variables, features, and fault types.

6. Conclusions and Discussion

This paper presents a classification method, based on feature sub-space neighbor vote, for small sample data of high dimensions from nonlinear and non-Gaussian time series. The performance of the proposed method is evaluated using a well-known dataset of the semiconductor metal etch process.

The addition of the FD feature to the magnitude and entropy of the relative power over a frequency range can extract a set of fingerprint features for accurate classification of the data of each fault type from those of other fault types. The fingerprint feature subspace can be used for the accurate identification of data from the same fault type. Since a feature subspace corresponds to a map of variable significance, this can easily interpret the effect of the fault on the sensing variables. Inversely, this significance map can aid in the cause analysis of significant or abnormal feature values of each variable.

The proposed method shows a promising solution to fault classification and feature description of a small nonlinear and non-Gaussian fault dataset. The proposed classification method uses all available fault data and their associated fault features in a neighbor voting for new fault data classification. Such a voting process only includes a similarity computation for each fault datum. It avoids the construction and computation of any accurate arbitrary mathematical model, such as the optimization calculation in FDA and SVM, which significantly relies on the amount of labeled data or the assumptions of the data distribution.

Using the proposed feature subspace as a fingerprint of a specific fault can easily and accurately classify, as well as identify, the fault from other fault types. Such a feature subspace can be extracted for any fault data of other manufacturing processes. Since each fault type has a unique fingerprint of

variables significance map, the matching of its data to the feature subspace has the minimum distance. To improve the fault classification accuracy, the suitable features may be defined according to the known properties of the process data, e.g., the nonlinear non-Gaussian data can be defined by the three types of features in this paper. Moreover, with more new fault data being identified, the feature subspace for each fault type will be more constrained and accurate for fault classification. Additionally, a priori knowledge of the manufacturing processes, sensor physics, and fault types can be integrated to select the appropriate features for fault data.

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

Funding: This research was funded in part by the National Science Foundation (grant no. CMMI1916866).

Acknowledgments: This work is supported in part by the National Science Foundation (grant no. CMMI1916866). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

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

References

1. Wise, B.M.; Gallagher, N.B.; Butler, S.W.; White, D.D.; Barna, G.G. A Comparison of Principal Component Analysis, Multiway Principal Component Analysis, Trilinear Decomposition and Parallel Factor Analysis for Fault Detection in a Semiconductor Etch Process. *J. Chemom.* **1999**, *13*, 379–396. [CrossRef]
2. Du, X. Fault Detection Using Bispectral Features and One-Class Classifiers. *J. Process Control* **2019**, *83*, 1–10. [CrossRef]
3. He, Q.P.; Wang, J. Fault Detection Using the K-Nearest Neighbor Rule for Semiconductor Manufacturing Processes. *IEEE Trans. Semicond. Manuf.* **2007**, *20*, 345–354. [CrossRef]
4. Zhou, Z.; Wen, C.; Yang, C. Fault Isolation Based on K-Nearest Neighbor Rule for Industrial Processes. *IEEE Trans. Ind. Electron.* **2016**, *63*, 2578–2586. [CrossRef]
5. Mahadevan, S.; Shah, S.L. Fault Detection and Diagnosis in Process Data Using One-Class Support Vector Machines. *J. Process Control* **2009**, *19*, 1627–1639. [CrossRef]
6. Wang, T.; Qiao, M.; Zhang, M.; Yang, Y.; Snoussi, H. Data-Driven Prognostic Method Based on Self-Supervised Learning Approaches for Fault Detection. *J. Intell. Manuf.* **2020**, *31*, 1611–1619. [CrossRef]
7. Yu, J.; Qin, S.J. Multiway Gaussian Mixture Model Based Multiphase Batch Process Monitoring. *Ind. Eng. Chem. Res.* **2009**, *48*, 8585–8594. [CrossRef]
8. Yu, J. Fault Detection Using Principal Components-Based Gaussian Mixture Model for Semiconductor Manufacturing Processes. *IEEE Trans. Semicond. Manuf.* **2011**, *24*, 432–444. [CrossRef]
9. Beyer, K.; Goldstein, J.; Ramakrishnan, R.; Shaft, U. When Is "Nearest Neighbor" Meaningful? In *Database Theory—ICDT'99*; Beeri, C., Buneman, P., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 1999; pp. 217–235. [CrossRef]
10. Zimek, A.; Schubert, E.; Kriegel, H.-P. A Survey on Unsupervised Outlier Detection in High-Dimensional Numerical Data. *Stat. Anal. Data Min. ASA Data Sci. J.* **2012**, *5*, 363–387. [CrossRef]
11. Fu, Y.; Gao, Z.; Liu, Y.; Zhang, A.; Yin, X. Actuator and Sensor Fault Classification for Wind Turbine Systems Based on Fast Fourier Transform and Uncorrelated Multi-Linear Principal Component Analysis Techniques. *Processes* **2020**, *8*, 1066. [CrossRef]
12. Lee, J.-M.; Yoo, C.; Lee, I.-B. Fault Detection of Batch Processes Using Multiway Kernel Principal Component Analysis. *Comput. Chem. Eng.* **2004**, *28*, 1837–1847. [CrossRef]
13. Deng, X.; Tian, X. Nonlinear Process Fault Pattern Recognition Using Statistics Kernel PCA Similarity Factor. *Neurocomputing* **2013**, *121*, 298–308. [CrossRef]
14. 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]

15. Fazai, R.; Ben Abdellafou, K.; Said, M.; Taouali, O. Online Fault Detection and Isolation of an AIR Quality Monitoring Network Based on Machine Learning and Metaheuristic Methods. *Int. J. Adv. Manuf. Technol.* **2018**, *99*, 2789–2802. [CrossRef]
16. Alcala, C.F.; Qin, S.J. Reconstruction-Based Contribution for Process Monitoring with Kernel Principal Component Analysis. In Proceedings of the 2010 American Control Conference, Baltimore, MD, USA, 30 June–2 July 2010; pp. 7022–7027. [CrossRef]
17. Deng, X.; Cai, P.; Cao, Y.; Wang, P. Two-Step Localized Kernel Principal Component Analysis Based Incipient Fault Diagnosis for Nonlinear Industrial Processes. *Ind. Eng. Chem. Res.* **2020**, *59*, 5956–5968. [CrossRef]
18. Lee, W.J.; Mendis, G.P.; Triebe, M.J.; Sutherland, J.W. Monitoring of a machining process using kernel principal component analysis and kernel density estimation. *J. Intell. Manuf.* **2020**, *31*, 1175–1189. [CrossRef]
19. Goodlin, B.E.; Boning, D.S.; Sawin, H.H.; Wise, B.M. Simultaneous Fault Detection and Classification for Semiconductor Manufacturing Tools. *J. Electrochem. Soc.* **2003**, *150*, G778. [CrossRef]
20. Verron, S.; Tiplica, T.; Kobi, A. Fault Detection and Identification with a New Feature Selection Based on Mutual Information. *J. Process Control* **2008**, *18*, 479–490. [CrossRef]
21. Fuente, M.J.; Garcia, G.; Sainz, G.I. Fault Diagnosis in a Plant Using Fisher Discriminant Analysis. In Proceedings of the 2008 16th Mediterranean Conference on Control and Automation, Ajaccio, France, 25–27 June 2008; pp. 53–58. [CrossRef]
22. Lu, W.-P.; Yan, X.-F. Visual Monitoring of Industrial Operation States Based on Kernel Fisher Vector and Self-organizing Map Networks. *Int. J. Control Autom. Syst.* **2019**, *17*, 1535–1546. [CrossRef]
23. Van, M.; Kang, H.-J. Bearing Defect Classification Based on Individual Wavelet Local Fisher Discriminant Analysis with Particle Swarm Optimization. *IEEE Trans. Ind. Inform.* **2016**, *12*, 124–135. [CrossRef]
24. Ge, Z.; Zhong, S.; Zhang, Y. Semisupervised Kernel Learning for FDA Model and Its Application for Fault Classification in Industrial Processes. *IEEE Trans. Ind. Inform.* **2016**, *12*, 1403–1411. [CrossRef]
25. Adil, M.; Abid, M.; Khan, A.Q.; Mustafa, G.; Ahmed, N. Exponential Discriminant Analysis for Fault Diagnosis. *Neurocomputing* **2016**, *171*, 1344–1353. [CrossRef]
26. Yin, Z.; Hou, J. Recent Advances on SVM Based Fault Diagnosis and Process Monitoring in Complicated Industrial Processes. *Neurocomputing* **2016**, *174*, 643–650. [CrossRef]
27. Cho, S.; Jiang, J. A Fault Detection and Isolation Technique Using Nonlinear Support Vectors Dichotomizing Multi-Class Parity Space Residuals. *J. Process Control* **2019**, *82*, 31–43. [CrossRef]
28. Jan, S.U.; Lee, Y.-D.; Shin, J.; Koo, I. Sensor Fault Classification Based on Support Vector Machine and Statistical Time-Domain Features. *IEEE Access* **2017**, *5*, 8682–8690. [CrossRef]
29. Yang, J.; Zhang, Y.; Zhu, Y. Intelligent Fault Diagnosis of Rolling Element Bearing Based on SVMs and Fractal Dimension. *Mech. Syst. Signal Process.* **2007**, *21*, 2012–2024. [CrossRef]
30. Zhang, Z.; Wu, J.; Ma, J.; Wang, X.; Zhou, C. Fault Diagnosis for Rolling Bearing Based on Lifting Wavelet and Morphological Fractal Dimension. In Proceedings of the 27th Chinese Control and Decision Conference 2015 CCDC, Qingdao, China, 23–25 May 2015; pp. 6351–6354. [CrossRef]
31. Li, X.; Yang, Y.; Pan, H.; Cheng, J.; Cheng, J. A novel deep stacking least squares support vector machine for rolling bearing fault diagnosis. *Comput. Ind.* **2019**, *110*, 36–47. [CrossRef]
32. Aoyagi, K.; Wang, H.; Sudo, H.; Chiba, A. Simple method to construct process maps for additive manufacturing using a support vector machine. *Addit. Manuf.* **2019**, *27*, 353–362. [CrossRef]
33. Jamil, F.; Abid, M.; Adil, M.; Haq, I.; Khan, A.Q.; Khan, S.F. Kernel Approaches for Fault Detection and Classification in PARR-2. *J. Process Control* **2018**, *64*, 1–6. [CrossRef]
34. Liu, Y.; Zeng, J.; Xie, L.; Luo, S.; Su, H. Structured Joint Sparse Principal Component Analysis for Fault Detection and Isolation. *IEEE Trans. Ind. Inform.* **2019**, *15*, 2721–2731. [CrossRef]
35. Shi, F.; Cao, H.; Zhang, X.; Chen, X. A Reinforced k-Nearest Neighbors Method with Application to Chatter Identification in High-Speed Milling. *IEEE Trans. Ind. Electron.* **2020**, *67*, 10844–10855. [CrossRef]
36. Sun, Z.; Yang, J.; Zheng, K. A Novel Fault Detection Method for Semiconductor Manufacturing Processes. In Proceedings of the 2019 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Auckland, New Zealand, 20–23 May 2019; pp. 1–6.
37. Ali, M.Z.; Shabbir, M.N.S.K.; Liang, X.; Zhang, Y.; Hu, T. Machine Learning-Based Fault Diagnosis for Single- and Multi-Faults in Induction Motors Using Measured Stator Currents and Vibration Signals. *IEEE Trans. Ind. Appl.* **2019**, *55*, 2378–2391. [CrossRef]

38. Han, T.; Jiang, D.; Zhao, Q.; Wang, L.; Yin, K. Comparison of Random Forest, Artificial Neural Networks and Support Vector Machine for Intelligent Diagnosis of Rotating Machinery. *Trans. Inst. Meas. Control* **2018**, *40*, 2681–2693. [CrossRef]
39. Cheng, Y.; Church, G.M. Biclustering of Expression Data. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* **2000**, *8*, 93–103. [PubMed]
40. Busygin, S.; Prokopyev, O.; Pardalos, P.M. Biclustering in Data Mining. *Comput. Oper. Res.* **2008**, *35*, 2964–2987. [CrossRef]
41. Wang, W.J.; Wu, Z.T.; Chen, J. Fault Identification in Rotating Machinery Using the Correlation Dimension and Bispectra. *Nonlinear Dyn.* **2001**, *25*, 383–393. [CrossRef]
42. Dong, G.; Chen, J.; Zhao, F. A Frequency-Shifted Bispectrum for Rolling Element Bearing Diagnosis. *J. Sound Vib.* **2015**, *339*, 396–418. [CrossRef]
43. Tian, J.; Morillo, C.; Azarian, M.H.; Pecht, M. Motor Bearing Fault Detection Using Spectral Kurtosis-Based Feature Extraction Coupled With K-Nearest Neighbor Distance Analysis. *IEEE Trans. Ind. Electron.* **2016**, *63*, 1793–1803. [CrossRef]
44. Li, Y.; Liang, X.; Zuo, M.J. Diagonal Slice Spectrum Assisted Optimal Scale Morphological Filter for Rolling Element Bearing Fault Diagnosis. *Mech. Syst. Signal Process.* **2017**, *85*, 146–161. [CrossRef]
45. He, S.; Chen, J.; Zhou, Z.; Zi, Y.; Wang, Y.; Wang, X. Multifractal Entropy Based Adaptive Multiwavelet Construction and Its Application for Mechanical Compound-Fault Diagnosis. *Mech. Syst. Signal Process.* **2016**, *76–77*, 742–758. [CrossRef]
46. Higuchi, T. Approach to an Irregular Time Series on the Basis of the Fractal Theory. *Phys. Nonlinear Phenom.* **1988**, *31*, 277–283. [CrossRef]
47. Accardo, A.; Affinito, M.; Carrozzi, M.; Bouquet, F. Use of the Fractal Dimension for the Analysis of Electroencephalographic Time Series. *Biol. Cybern.* **1997**, *77*, 339–350. [CrossRef] [PubMed]
48. Caesarendra, W.; Kosasih, B.; Tieu, K.; Moodie, C.A.S. An Application of Nonlinear Feature Extraction—A Case Study for Low Speed Slewing Bearing Condition Monitoring and Prognosis. In Proceedings of the 2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Wollongong, NSW, Australia, 9–12 July 2013; pp. 1713–1718. [CrossRef]
49. Ge, Z.; Gao, F.; Song, Z. Batch process monitoring based on support vector data description method. *J. Process Control* **2011**, *21*, 949–959. [CrossRef]