






Review

# A Review of Kernel Methods for Feature Extraction in Nonlinear Process Monitoring

Karl Ezra Pilario <sup>1,2,\*</sup> , Mahmood Shafiee <sup>1,3,\*</sup> , Yi Cao <sup>4,\*</sup> , Liyun Lao <sup>1</sup>  and Shuang-Hua Yang <sup>4</sup> 

<sup>1</sup> Department of Energy and Power, Cranfield University, Bedfordshire MK43 0AL, UK; l.lao@cranfield.ac.uk

<sup>2</sup> Department of Chemical Engineering, University of the Philippines Diliman, Quezon City 1101, Philippines

<sup>3</sup> School of Engineering and Digital Arts, University of Kent, Canterbury CT2 7NT, UK

<sup>4</sup> College of Chemical and Biological Engineering, Zhejiang University, Hangzhou 310027, China; yangsh@zju.edu.cn

\* Correspondence: k.pilario@cranfield.ac.uk (K.E.P.); m.shafiee@cranfield.ac.uk (M.S.); caoyi2018@zju.edu.cn (Y.C.)

Received: 22 November 2019; Accepted: 15 December 2019; Published: 23 December 2019



**Abstract:** Kernel methods are a class of learning machines for the fast recognition of nonlinear patterns in any data set. In this paper, the applications of kernel methods for feature extraction in industrial process monitoring are systematically reviewed. First, we describe the reasons for using kernel methods and contextualize them among other machine learning tools. Second, by reviewing a total of 230 papers, this work has identified 12 major issues surrounding the use of kernel methods for nonlinear feature extraction. Each issue was discussed as to why they are important and how they were addressed through the years by many researchers. We also present a breakdown of the commonly used kernel functions, parameter selection routes, and case studies. Lastly, this review provides an outlook into the future of kernel-based process monitoring, which can hopefully instigate more advanced yet practical solutions in the process industries.

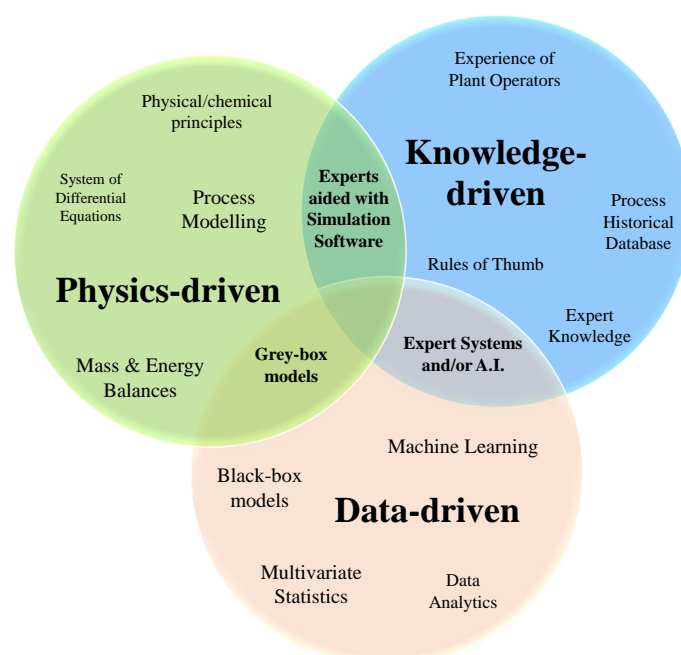
**Keywords:** kernel PCA; kernel PLS; kernel ICA; kernel CCA; kernel CVA; kernel FDA; multivariate statistics; fault detection; fault diagnosis; machine learning

## 1. Introduction

Process monitoring refers to various methods used for the detection, diagnosis, and prognosis of faults in industrial plants [1,2]. In literature, the term “fault” has been defined as any unpermitted deviation of at least one process parameter or variable in the plant [3]. Although controls are already in place to compensate for process upsets and disturbances, process faults can still occur [1]. These faults include sensor faults (e.g., measurement bias), actuator faults (e.g., valve stiction), fouling, loss of material, drifting reaction kinetics, pipe blockages, etc. Fault detection, diagnosis, and prognosis methods aim to, respectively, determine the presence, identify the cause, and predict the future behavior of these process anomalies [2,4]. Thus, process monitoring is a key layer of safety for maintaining an efficient and reliable operation of industrial plants [5].

In general, process monitoring can be performed using either a physics-driven, knowledge-driven, or data-driven approach (see Figure 1) [1,6]. Among these, the data-driven approach may be preferred due to the following reasons. Physics-driven methods rely on a first-principles model of the system, i.e., mass-and-energy balances and physical/chemical principles, which is used to check how well the theory agrees with the observed plant data. However, these models are difficult to construct given the complexity of modern industrial plants [6]. Similarly, knowledge-driven methods rely on expert knowledge and the experience of plant operators to judge process conditions, but a comprehensive

knowledge base may be too time-consuming to accumulate and codify precisely [6]. In contrast, data-driven methods rely only on plant data from which statistical models can be built to classify normal from faulty conditions. Nowadays, plant data sets are generated in abundance [7]. Samples are collected from online sensors on hundreds to thousands of process variables every few seconds [8] via Supervisory Control and Data Acquisition (SCADA) systems. Many researchers have long recognized the opportunity to exploit these data sets for process monitoring, and this led to the development of Multivariate Statistical Process Monitoring (MSPM) methods. Data-driven methods and MSPM provide the context to this review paper. However, in the larger context, process monitoring researchers must still aim for the right synergy between physics-, knowledge-, and data-driven technologies.



**Figure 1.** Three categories of process monitoring methods. See [1,6] for more details.

The popularity of data-driven MSPM methods has increased in the past few decades, especially towards the advent of the Industry 4.0 era. Applications of machine learning [9–11], Big Data [12,13], artificial intelligence (AI) [14], and process data analytics [15,16] to the process systems engineering (PSE) field are now gaining acceptance. Deep neural nets, support vector machines, fuzzy systems, principal components analysis,  $k$ -nearest neighbors, K-means clustering, etc., are now being deployed to analyze plant data, generate useful information, and translate results into key operational decisions. For instance, Patwardhan et al. [17] recently reported real-world applications of these methods for predictive maintenance, alarm analytics, image analytics, and control performance monitoring, among others. Applications of the MSPM methods to an industrial-scale multiphase flow facility at Cranfield University have also been reported in [18,19]. Until now, new methods are still being developed within the machine learning and AI community, and so do their applications in PSE. This means that it may be difficult to select which data-driven methods to use. Nevertheless, chemical engineers can apply their domain expertise to match the right solutions to the right engineering problems.

Despite the benefits of data-driven techniques, it is still challenging to use them for process monitoring due to many issues that arise in practice. One key issue that is highlighted in this paper is the fact that real-world systems are nonlinear [20]. More precisely, the relationship between the process variables are nonlinear. For example, pressure drop and flow rate have a squared relationship according to Bernoulli's equation, outlet stream temperature and composition in a chemical reactor are nonlinearly related due to complex reaction kinetics, and so on. These patterns must be learned

and taken into account in the statistical models. If the analysis of data involves linear methods alone, fault detection may be inaccurate, yielding many false alarms and missed alarms. Note, however, that linear methods can still be applied provided that the plant conditions are kept sufficiently close to a single operating point. This is due to the fact that a first-degree (linear) Taylor series approximation of the variable relationships can be assumed close to a fixed point. Linear methods are attractive because they rely only on simple linear algebra and matrix theory, which are elegant and computationally accessible. However, if the plant is operating at a wide range of conditions, the resulting nonlinear dynamic behavior must be addressed with more advanced techniques.

*Kernel methods or kernel machines* are a class of machine learning methods that can be used to handle the nonlinear issue. The main idea behind kernel methods is to pre-process the data by projecting them onto higher-dimensional spaces where linear methods are more likely to be applicable [21]. Thus, kernel methods can discover nonlinear patterns from the data while retaining the computational elegance of matrix algebra [22]. In the process monitoring context, kernel learning is mostly used in the *feature extraction* step of the analysis of plant data. In this paper, we review the applications of kernel methods for feature extraction in nonlinear process monitoring.

In detail, the objectives of this review are: (1) To motivate the use of kernel methods for process monitoring; (2) To identify the issues regarding the use of kernel methods to perform feature extraction for nonlinear process monitoring; (3) To review the literature on how these issues were addressed by researchers; and (4) To suggest future research directions on kernel-based process monitoring. This work is mainly dedicated to the review of kernel-based process monitoring methods, which has not appeared before to the best of the authors' knowledge. Other related reviews that may be of interest to the reader are also available, as listed Table 1, along with their relationship to this paper.

This review paper is timely for two reasons. The original proponent of the first developed kernel feature learner called *kernel principal components analysis (KPCA)* was Bernhard Schölkopf [22] in a 1998 paper, together with Alexander Smola and Klaus-Robert Müller. KPCA paved the framework for more kernel extensions of linear machines, known today as kernel methods. For his contributions, Schölkopf was awarded the Körber Prize last September 2019, which is "the scientific distinction with the highest prize money in Germany" [23]. This recognition highlights the impact kernel methods have made to the field of data analytics. The purpose of this paper is to showcase this impact in the process monitoring field. Shortly after, Lee et al. [24] was the first to use KPCA for nonlinear process monitoring in 2004. Hence, this paper is timely as it reviews the development of kernel-based process monitoring research for the last 15 years since the first application by Lee et al.

This paper is organized as follows. In Section 2, we first motivate the use of kernel methods and situate them among other machine learning tools. Section 3 provides the methodology on how the literature review was conducted, and also includes a brief summary of review results. The main body of this paper is Section 4, where we detail the issues surrounding the use of kernel methods in practice, and the many ways researchers have addressed them through the years. A future outlook on this area of research is given in Section 5. Finally, the paper is concluded in Section 6.

**Table 1.** Other recent reviews and their relationship to the present review.

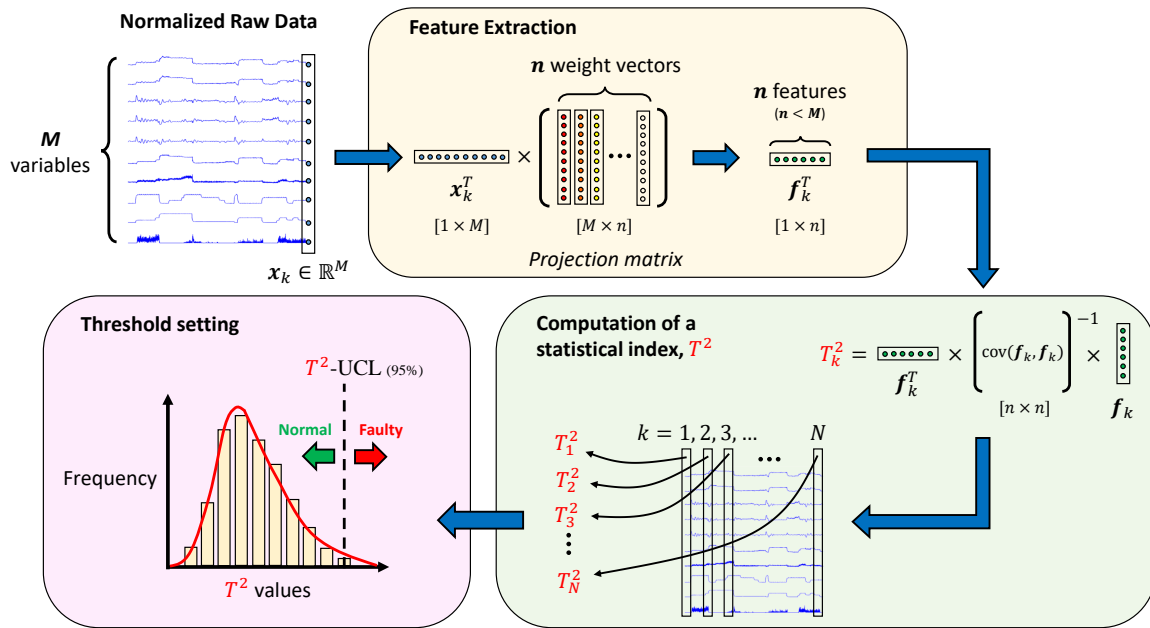
Year	Reference	Remark
2012	Qin [25]	Discusses the general issues and explains how basic data-driven process monitoring (MSPM) methods work.
2012	MacGregor and Cinar [26]	Reviews data-driven models not only in process monitoring, but also in optimization and control.
2013	Ge et al. [6]	Reviews data-driven process monitoring using recent MSPM tools and discusses more recent issues.
2014	Yin et al. [27]	Reviews data-driven process monitoring but from an application point of view; it also provides a basic monitoring framework.
2014	Ding et al. [28]	Reviews data-driven process monitoring methods with specific focus on dynamic processes.
2014	Qin [15]	Gives an overview of process data analytics, in which process monitoring is only one of the applications.
2015	Yin et al. [29]	Reviews data-driven methods not only in industrial processes, but also in smart grids, energy, and power systems, etc.
2015	Severson et al. [30]	Gives an overview of process monitoring in a larger context than just data-driven methods, and advocates hybrid methods.
2016	Tidriri et al. [31]	Compares physics-driven and data-driven process monitoring methods, and reviews recent hybrid approaches.
2016	Yin and Hou [32]	Reviews process monitoring methods that used support vector machines (SVM) for electro-mechanical systems.
2017	Lee et al. [9]	Reviews recent progresses and implications of machine learning to the field of PSE.
2017	Ge et al. [11]	Reviews data-driven methods in the process industries from the point of view of machine learning.
2017	Ge [33]	Reviews data-driven process monitoring methods with specific focus on dealing with the issues on the plant-wide scale.
2017	Wang et al. [34]	Reviews MSPM algorithms from 2008 to 2017, including both papers and patents in Web of Science, IEEE Xplore, and the China National Knowledge Infrastructure (CNKI) databases.
2018	Md Nor et al. [35]	Reviews data-driven process monitoring methods with guidelines for choosing which MSPM and machine learning tools to use.
2018	Alauddin et al. [36]	Gives a bibliometric review and analysis of the literature on data-driven process monitoring.
2019	Qin and Chiang [16]	Reviews machine learning and AI in PSE and advocates the integration of data analytics to chemical engineering curricula.
2019	Jiang et al. [37]	Reviews data-driven process monitoring methods with specific focus on distributed MSPM tools for plant-wide monitoring.
2019	Quiñones-Grueiro et al. [38]	Reviews data-driven process monitoring methods with specific focus on handling the multi-mode issue.
	This paper	Reviews data-driven process monitoring methods that applied kernel methods for feature extraction.

## 2. Motivation for Using Kernel Methods

To motivate the use of kernel methods, we first discuss how a typical data-driven fault detection framework works (see Figure 2). A plant data set for model training usually consists of  $N$  samples of  $M$  variables collected at normal operating conditions. This data is normalized so that the analysis is unbiased to any one variable, i.e., all variables are treated equally. Firstly, the data set undergoes a feature extraction step. We refer to *feature extraction* as any method of transforming the data in order to reveal a reduced set of mutually independent signals, called *features*, that are most sensitive to process faults. In Figure 2, this step is carried out by multiplying a projection matrix of weight vectors to a vector of samples,  $\mathbf{x}_k$ , at the  $k$ th instant. Secondly, a statistical index is built from the features, which serves as a health indicator of the process. The most commonly used index is Hotelling's  $T^2$ , which is computed as shown in the figure as well. Finally, the actual anomaly detector is trained by analyzing the distribution of  $T^2$ . In this step, the aim is to find an upper bound or threshold on the normal  $T^2$

values, called the *upper control limit* or UCL. This threshold is based on a user-defined confidence level, e.g., 95%, which represents the fraction of the area under the distribution of  $T^2$  that is below the UCL. During the online phase, an alarm is triggered whenever the computed  $T^2$  exceeds the fixed  $T_{UCL}^2$ , signifying the presence of a fault.

When a fault is detected, fault diagnosis is usually achieved by identifying the variables with the largest contributions to the value of  $T^2$  at that instant. Lastly, fault prognosis can be performed by predicting the future evolution of the faulty variables or the  $T^2$  index itself.



**Figure 2.** Basic steps of typical Multivariate Statistical Process Monitoring (MSPM) methods to achieve fault detection. Here, the feature extraction step shows only a linear transformation of data.

### 2.1. Feature Extraction Using Kernel Methods

Among the three basic steps in Figure 2, feature extraction is found to have the greatest impact to process monitoring performance. Even in other contexts, feature engineering is regarded as the one aspect of machine learning that is domain-specific and, hence, requires creativity from the user [39,40]. As such, traditional MSPM methods mainly differ in how the weight vectors are obtained. Weights can be computed via principal components analysis (PCA), partial least squares (PLS), independent components analysis (ICA), Fisher/linear discriminant analysis (FDA or LDA), or canonical correlation analysis (CCA) [1]. However, only a linear transformation of the data is involved in these methods. Mathematically, a linear transformation can be written as:

$$\mathbf{f}_k = \mathbf{W}_n^T \mathbf{x}_k, \quad (1)$$

where  $\mathbf{W}_n \in \mathbb{R}^{M \times n}$  is the projection matrix,  $\mathbf{f}_k \in \mathbb{R}^n$  are the features, and  $\mathbf{x}_k \in \mathbb{R}^M$  are the normalized raw data at the  $k$ th instant. For the case of PCA, the  $\mathbf{W}$  can be computed by diagonalizing the sample covariance matrix,  $\mathbf{C} = \text{cov}(\mathbf{x}_k, \mathbf{x}_k)$ , as [1]:

$$\mathbf{C} = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T \in \mathbb{R}^{M \times M}, \quad (2)$$

$$\mathbf{W} = \mathbf{V}^T \mathbf{\Lambda}^{-1/2} \in \mathbb{R}^{M \times M}, \quad (3)$$

where  $\mathbf{V}$  contains the eigenvectors with corresponding eigenvalues in  $\mathbf{\Lambda}$ . Only the first  $n$  columns of  $\mathbf{W}$  are taken to finally yield  $\mathbf{W}_n$ . The weights from PCA are orthogonal basis vectors that describe directions of maximum variance in the data set [1].

In order to generate nonlinear features, a nonlinear mapping can be used to transform the data,  $\phi(\mathbf{x})$ , so that Equation (1) becomes  $\mathbf{f}_k = \mathbf{W}_n^T \phi(\mathbf{x}_k)$ . However, the mapping  $\phi(\cdot)$  is unknown and difficult to design. In 1998, Schölkopf et al. [22] proposed to replace the sample covariance matrix,  $\mathbf{C} = \text{cov}(\phi(\mathbf{x}_k), \phi(\mathbf{x}_k))$ , by a kernel matrix  $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$  whose elements are computed by a *kernel function*,  $k(\cdot, \cdot)$ . They have shown that if the kernel function satisfies certain properties, it can act as a dot product in the feature space. That is, the  $\mathbf{K}_{ij}$  can take the role of a covariance matrix of nonlinear features. By adopting a kernel function, the need to specify  $\phi(\cdot)$  has now been avoided, and this realization has been termed as the *kernel trick* [22]. The result is a method called kernel principal components analysis (KPCA) [22], a nonlinear learner trained by merely solving the eigenvalue decomposition of  $\mathbf{K}_{ij}$  as in Equation (2). As mentioned in Section 1, KPCA is the first kernel method applied to process monitoring as a feature extractor [24].

Upon using kernel methods, the nonlinear transformation is now equivalent to [22]:

$$\mathbf{f}_k = \left[ \sum_{j=1}^N \mathbf{w}_i^T k(\mathbf{x}_k, \mathbf{x}'_j) \right]_{i=1, \dots, n} \quad (4)$$

where  $\mathbf{w}_i \in \mathbb{R}^M$  is a column weight vector,  $\mathbf{f}_k \in \mathbb{R}^n$  are the features,  $\mathbf{x}_k \in \mathbb{R}^M$  is the new data to be projected,  $\mathbf{x}' \in \mathbb{R}^M$  is the training data set, and  $k(\cdot, \cdot)$  is the kernel function. The kernel function is responsible for projecting the data onto high-dimensional spaces where, according to Cover's theorem [21], the features are more likely to be linearly separable. This high-dimensional space is known in functional analysis as a Reproducing Kernel Hilbert Space (RKHS) [22]. Usual choices of kernel functions found from this review are as follows:

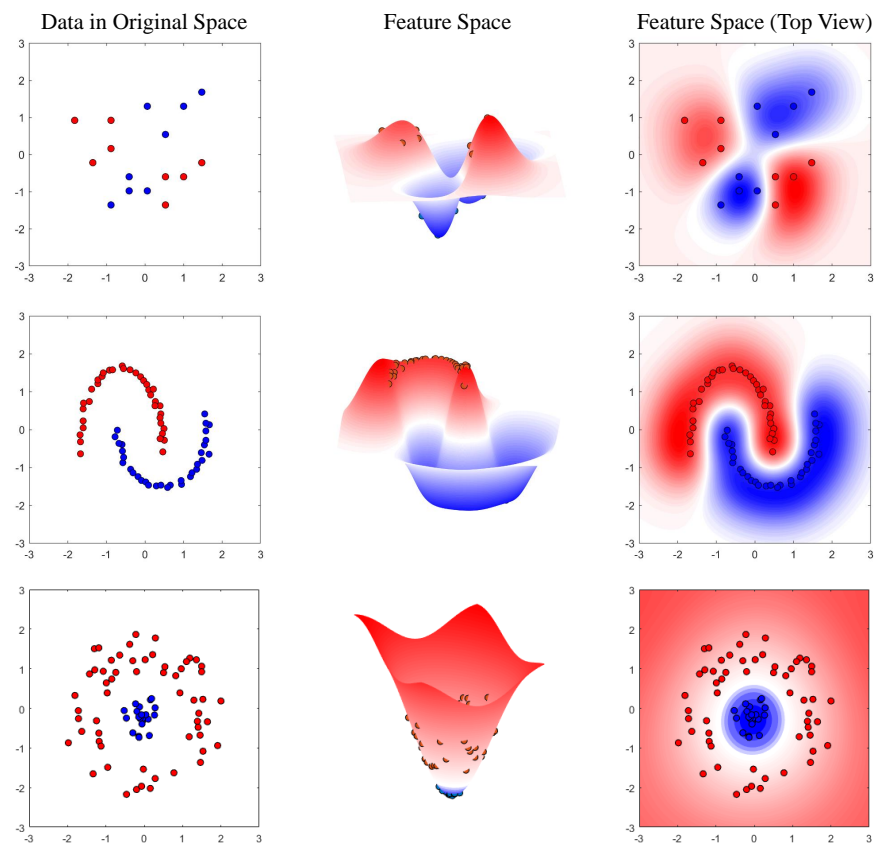
$$\text{Gaussian radial basis function (RBF):} \quad k(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{-\|\mathbf{x} - \mathbf{x}'\|^2}{c}\right) \quad (5)$$

$$\text{Polynomial kernel (POLY):} \quad k(\mathbf{x}, \mathbf{x}') = (\langle \mathbf{x}, \mathbf{x}' \rangle + 1)^d \quad (6)$$

$$\text{Sigmoid kernel (SIG):} \quad k(\mathbf{x}, \mathbf{x}') = \tanh(a\langle \mathbf{x}, \mathbf{x}' \rangle + b), \quad (7)$$

where  $a, b, c, d$  are kernel parameters to be determined by various selection routes.

To understand what happens in the kernel mapping, Figure 3 shows three sample data sets and their projections in the kernel feature space. The red and blue data points belong to different classes, and evidently, it is impossible to separate them by a straight line in the original data space. However, after a kernel transformation onto a higher dimensional space, it is now possible to separate them using a linear plane (white contour), which translates to a nonlinear boundary in the original space. In these examples, an RBF kernel of various  $c$  values was used, Equation (5), and the transformation is computed using Support Vector Machines (SVM). More theoretical details on kernel methods, KPCA, and SVM can be found in other articles [22,41,42], as well as books such as *Kernel Methods for Pattern Analysis* by Shawe-Taylor and Cristianini [43], *Support Vector Machines and Other Kernel-based Learning Methods* by Cristianini and Shawe-Taylor [44], and *Pattern Recognition and Machine Learning* by Bishop [45].



**Figure 3.** Illustration of kernel nonlinear transformation. These were generated with code available in <https://uk.mathworks.com/matlabcentral/fileexchange/65232-binary-and-multi-class-svm>.

## 2.2. Kernel Methods in the Machine Learning Context

Aside from kernel methods, other tools from machine learning can also be applied to process monitoring. Figure 4 gives an overview of learning methods that are relevant to process monitoring, from the authors' perspective. Each method in this figure represents a body of associated techniques, and so the reader can search using these keywords to learn more. More importantly, the methods that were marked with an asterisk (\*) have a "kernelized" version, and so they belong to the family of kernel methods. To kernelize means to apply the kernel trick to a previously linear machine. For example, PCA becomes Kernel PCA, Ridge Regression becomes Kernel Ridge Regression, K-means clustering becomes Kernel K-means, and so on. All these methods were developed to solve a particular learning problem or learning task, such as classification, regression, clustering, etc.

Supervised and unsupervised learning are the two main categories of learning tasks (although semi-supervised, reinforcement, and self-supervised learning categories also exist [9,11,46]). According to Murphy [47], learning is *supervised* if the goal is to learn a mapping from inputs to outputs, given a labeled set of input-output pairs. On the other hand, learning is *unsupervised* if the goal is to discover patterns from a data set without any label information. In the context of process monitoring, examples of learning problems under each category can be listed as follows:

- **Supervised learning**

*Classification:* Given data samples labeled as normal and faulty, find a boundary between the two classes; or, given samples from various fault types, find a boundary between the different types.

*Regression:* Given samples of regressors (e.g., process variables) and targets (e.g., key performance indicators), find a function of the former that predicts the latter; or, find a model for predicting the future evolution of process variables whether at normal or faulty conditions.

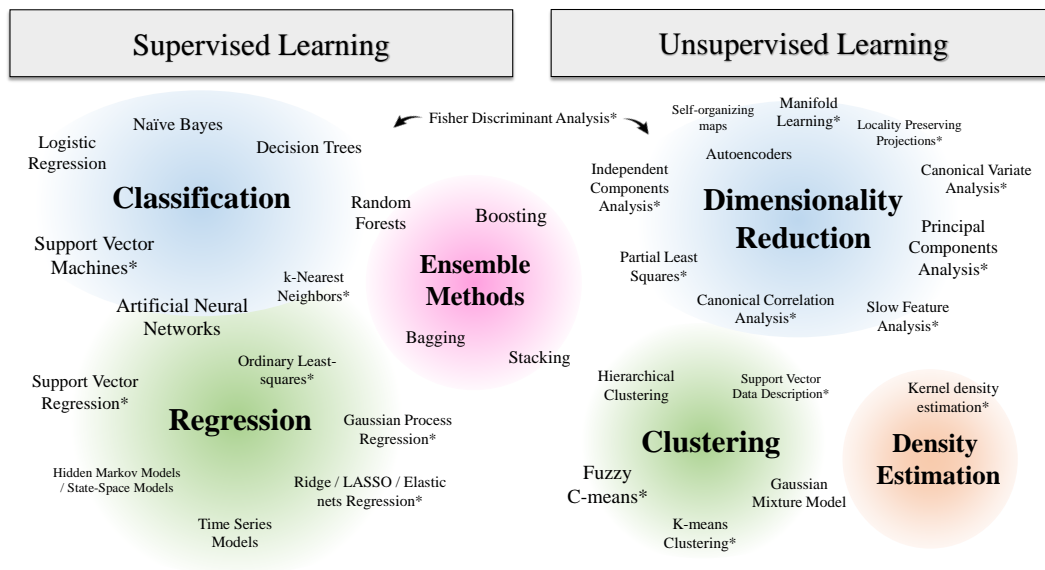
*Ensemble methods:* Find a strategy to combine results from several models.

- **Unsupervised learning**

*Dimensionality reduction:* Extract low-dimensional features from the original data set that can enable process monitoring or data visualization.

*Clustering:* Find groups of similar samples within the data set, without knowing beforehand whether they are normal or faulty.

*Density Estimation:* Find the probability distribution of the data set.



**Figure 4.** Machine learning methods relevant to process monitoring (from the authors' perspective). Those with (\*) have versions that belong to the family of kernel methods.

In relation to the framework in Figure 2, one possible correspondence would be the following: (1) Use dimensionality reduction or clustering for feature extraction; (2) Use density estimation for threshold setting; (3) Use classification for diagnosis; and, (4) Use regression for prognosis and other predictive tasks. It is clear from Figure 4 that kernel methods can participate in any stage of the process monitoring procedure, not just in the feature extraction step. In fact, many existing frameworks already used kernel support vector machines (SVM) for fault classification, kernel density estimation (KDE) for threshold setting, etc. We also note that many other alternatives to kernel methods can be used to perform each learning task. For instance, an early nonlinear extension of PCA for process monitoring was based on principal curves and artificial neural networks (ANN) by Dong and McAvoy [48] in 1996. Even today, ANNs are still a popular alternative to kernel methods.

### 2.3. Relationship between Kernel Methods and Neural Networks

Neural networks are attractive due to their universal approximation property [49], that is, they can theoretically approximate any function to an arbitrary degree of accuracy [45]. Both ANNs and kernel methods can be used for nonlinear process monitoring. However, one important difference between them is in the computational aspect. Kernel methods such as KPCA are faster to train (see Section 2.1), whereas ANNs require an iterative process for training (i.e., gradient descent) because of the need to solve a nonlinear optimization problem [44]. But during the online phase, kernel methods may be slower since they need to store a copy of the training data in order to make predictions for new test data (see Equation (4)) [45]. In ANNs, once the parameters have been learned, the training data set can be discarded [45]. Thus, kernel methods have issues with scalability. Another distinction is provided



by Pedro Domingos in his book *The Master Algorithm* [50] in terms of learning philosophy: If ANNs learn by mimicking the structure of the brain, kernel methods learn by analogy. Indeed, the reason why kernel methods need to store a copy of the training data is so that it can compute the similarity between any test sample and the training samples. The similarity measure is provided by the kernel function,  $k(\cdot, \cdot)$  [44]. However, selecting a kernel function is also a long-standing issue. Later on, this review includes a survey of the commonly used kernel functions for process monitoring.

Despite the many distinctions between kernel methods and ANNs, neither of them is clearly superior to the other. Presently, many of the drawbacks of each are already being addressed, and their unique benefits are also being enhanced. Also, these two approaches are connected in some ways, as explained in [45]. For instance, the nonlinear kernel transformation in Equation (4) can be interpreted as a two-layer network [51]: the first layer corresponds to  $\mathbf{x}_k \rightarrow k(\mathbf{x}_k, \mathbf{x}')$ , while the second layer corresponds to  $k(\mathbf{x}_k, \mathbf{x}') \rightarrow \mathbf{f}_k$  with weights,  $\mathbf{w}_i$ .

ANNs have found success in many areas, especially in computer vision where deep ANNs [52] have reportedly surpassed human-level performance for image recognition [53]. Opportunities for applying deep ANNs to the field of PSE were also given in [9]. Meanwhile, kernel methods were shown to have matched the accuracy of deep ANNs for speech recognition [54]. In the real world, kernel methods have been applied successfully to wind turbine performance assessment [55], machinery prognostics [56], and objective flow regime identification [57], to name a few.

In the AI community, methods that *combine* kernel methods with deep learning are now being developed, such as neural kernel networks [58,59], deep neural kernel blocks [60], and deep kernel learning [61,62]. A soft sensor based on deep kernel learning was recently applied in a polymerization process [63]. Based on these recent advances, Wilson et al. [62] has concluded that the relationship between kernel methods and deep ANNs must not be competing, but rather, complementary. Perhaps a more forward-looking claim would be that of Belkin et al. [51], who said that “in order to understand deep learning we need to understand kernel learning”. Therefore, kernel methods are powerful and important machine learning tools that are worthwhile to consider in practice.

### 3. Methodology and Results Summary

Having motivated the importance of kernel methods in the previous section, the rest of the paper is dedicated to a review of their applications to process monitoring.

#### 3.1. Methodology

The scope of this review is limited to the applications of kernel methods in the *feature extraction* step of process monitoring. This is because we are after the important issues in feature extraction that may drive future research directions. Papers that used kernelized MSPM tools such as kernel PCA, kernel ICA, kernel PLS, kernel FDA, kernel SFA, kernel CCA, kernel LPP, kernel CVA, etc. were included, although their details are not given here. Meanwhile, papers that used kernel methods in other stages of process monitoring (e.g., SVMs for fault classification, Gaussian Processes (GP) for fault prediction, and KDE for threshold setting) may also appear, but these are not the main focus. Moreover, this review only includes papers with industrial process case studies, such as the Tennessee Eastman Plant benchmark. A review of literature on the condition monitoring of electro-mechanical system case studies (e.g., rotating machinery) can be found elsewhere [64,65]. Interested practitioners are also referred to Wang et al. [34] for a survey of patents related to process monitoring.

For this review, an extensive literature search was conducted on the following journals: (1) IEEE Transactions on Industrial Informatics; (2) IEEE Transactions on Industrial Electronics; (3) IEEE Transactions on Control Systems Technology; (4) IEEE Transactions on Automation Science and Engineering; (5) IEEE Access; (6) Chemical Engineering Science; (7) Chemometrics and Intelligent Laboratory Systems; (8) Computers and Chemical Engineering; (9) Chemical Engineering Research and Design; (10) Journal of Process Control; (11) Control Engineering Practice; (12) ISA Transactions; (13) Expert Systems with Applications; (14) Chinese Journal of Chemical Engineering; (15) Industrial

and Engineering Chemistry Research; (16) Process Safety and Environmental Protection; (17) Journal of Chemometrics; (18) AIChE Journal; and, (19) Canadian Journal of Chemical Engineering. The keywords used for searching were “kernel and fault”. Keywords such as “monitoring”, “detection”, and “diagnosis” were not used because not all intended papers contain these words in the text. From the search results, only the papers that fit the aforementioned scope were included; 155 papers were found this way. Also, selected papers from other journals and conference proceedings were found by following citations forwards and backwards. However, a comprehensive search is not guaranteed. The entire search process was performed in October 2019, and hence, only published works until this time were found. In the end, a total of 230 papers were included in this review.

### 3.2. Results Summary

Figure 5 shows the distribution of the reviewed papers by year of publication. The overall increasing trend in the number of papers indicate that kernel-based feature extraction is being adopted by more and more process monitoring researchers. Figure 6a then shows the most commonly used kernelized feature extractors for nonlinear process monitoring. Kernel PCA is most widely used, followed by kernel PLS, kernel ICA, kernel FDA, kernel CVA, and so on. The widespread use of kernel PCA can be attributed to the fact that linear algorithms can be kernelized by performing kernel PCA followed by the linear algorithm itself. For instance, kernel ICA is equivalent to kernel PCA + ICA [66]. Likewise, kernel CVA can be performed as kernel PCA + CVA [67]. Hence, kernel PCA was cited more frequently than other techniques.

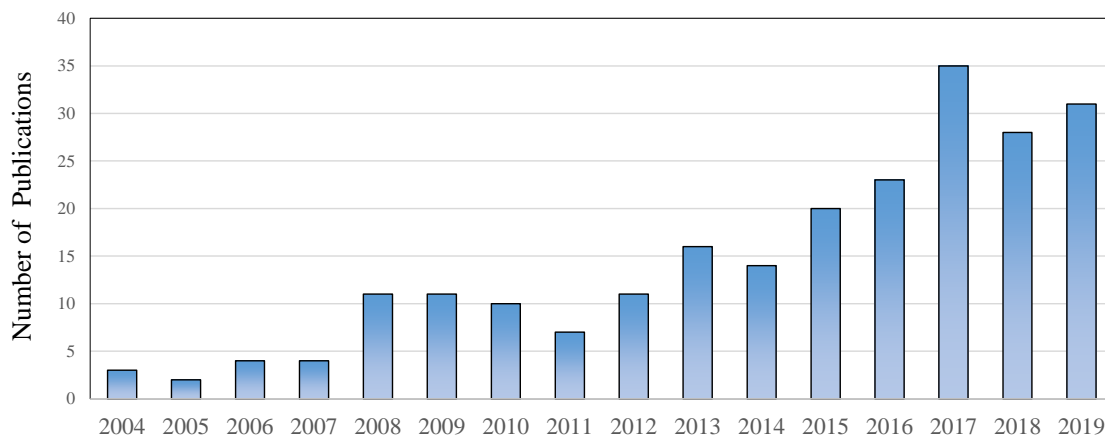


Figure 5. Yearly distribution of publications found in the literature review.

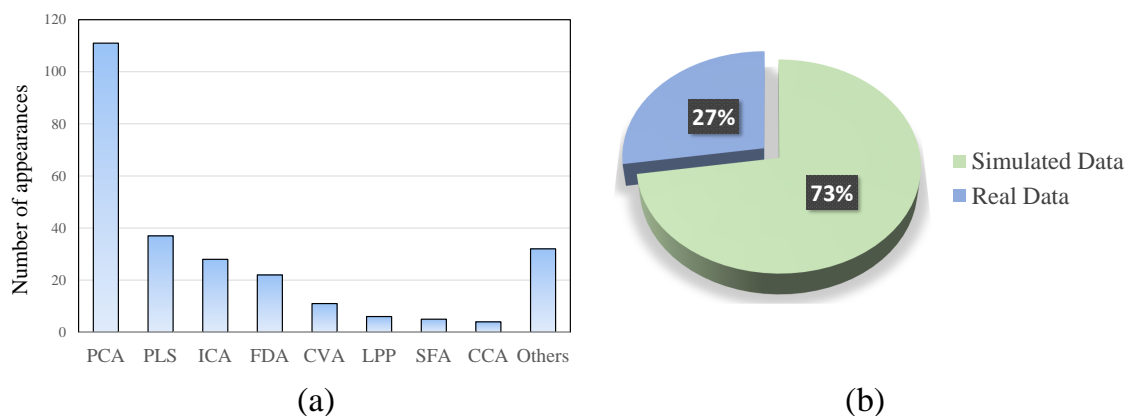


Figure 6. (a) Commonly used kernelized methods found in the review; (b) Breakdown of the type of case studies found in the review.

In the reviewed papers, application case studies were also used for evaluating the effectiveness of the proposed kernel methods for process monitoring. Figure 6b shows the breakdown of papers according to the type of case study they used: simulated or real-world. As shown, only 27% of the papers have indicated the use of at least one real-world data set, taken from either industrial processes or laboratory experiments. On the other hand, the rest of the papers used simulated data sets alone for testing. The Tennessee Eastman Plant (TEP) is found to be the most commonly used simulated case study. It may still be advantageous to use simulated case studies since the characteristics of the simulated data are usually known or can be built in the simulator. Hence, the user can highlight the strengths of a particular method by its ability to handle certain data characteristics. Another advantage of using simulated data is that tests can be repeated many times by performing many Monte Carlo simulations. Nevertheless, the ultimate goal should still be to assess the proposed methods on real-world data. For instance, in a paper by Fu et al. [68], kernel PCA and kernel PLS were applied to 3 different real-world data sets: two from the chemical process industry and one from a laboratory mixing experiment. Among the chemical processes is a butane distillation system. Vitale et al. [69] also used real-world data sets from the pharmaceutical industry to test kernel methods. Results from these examples have proven that handling the nonlinear issue is important for monitoring real-world industrial processes.

However, issues arise in the application of kernel methods for nonlinear process monitoring. After a careful study of the papers, 12 major issues were identified and listed in Table 2. The table includes the number of papers that addressed each of them. Although some of these issues are not unique to kernel methods alone, we review them within the context of kernel-based feature extraction. The bulk of this paper is dedicated to the discussion of these issues.

A list of all the reviewed papers is then given in Table 3. The table also shows the kernelized method they used, the case studies they used, the kernel functions they used, and more importantly, the issues they addressed. The purpose of this table is to help the reader choose a specific issue of interest (A to L) and peruse down the column for papers that addressed it. In the column on case studies, we have also highlighted in bold the ones that are real-world or industrial applications. The reader is referred to the appendix for the list of all abbreviations in this table.

**Table 2.** Issues surrounding the use of kernel methods for process monitoring.

Label	Name of Issue	No. of Papers That Addressed It
A	Batch process monitoring	30
B	Dynamics, multi-scale, and multi-mode monitoring	72
C	Fault diagnosis in the kernel feature space	100
D	Handling non-Gaussian noise and outliers	41
E	Improved sensitivity and incipient fault detection	39
F	Quality-relevant monitoring	37
G	Kernel design and kernel parameter selection	30
H	Fast computation of kernel features	34
I	Manifold learning and local structure analysis	20
J	Time-varying behavior and adaptive kernel computation	26
K	Multi-block and distributed monitoring	15
L	Advanced methods: Ensembles and Deep Learning	8

**Table 3.** Summary of papers: The issues they addressed and the kernel method, case studies, and kernel functions they used.

Year	Reference	Kernelized Method/s	Issues Addressed												Case Studies	Kernel/s Used		
			A	B	C	D	E	F	G	H	I	J	K	L				
1	2004	Lee et al. [24]	PCA	First application												NE, WWTP	RBF	
2	2004	Lee et al. [71]	PCA	✓													PenSim	POLY
3	2004	Choi and Lee [85]	PCA		✓												NE, WWTP	RBF
4	2005	Choi et al. [86]	PCA			✓											NE, CSTR	RBF
5	2005	Cho et al. [87]	PCA			✓											NE, CSTR	RBF
6	2006	Yoo and Lee [88]	PCA		✓			✓									NE, WWTP	RBF
7	2006	Lee et al. [89]	PCA, PLS							✓							<b>BAFP</b>	RBF
8	2006	Zhang et al. [90]	ICA				✓										FCCU	-
9	2006	Deng and Tian [91]	PCA		✓	✓											CSTR	RBF
10	2007	Zhang and Qin [72]	PCA, ICA	✓			✓										NPP	RBF
11	2007	Cho [74]	FDA	✓		✓											PCBP, PenSim	POLY
12	2007	Cho [92]	FDA			✓											TEP	RBF
13	2007	Sun et al. [93]	PCA			✓				✓							NE, <b>Rot. Machines</b>	RBF
14	2008	Choi et al. [94]	PCA		✓	✓											CSTR	RBF
15	2008	Tian and Deng [95]	PCA		✓	✓											TEP	RBF
16	2008	Wang et al. [96]	PCA	✓	✓												NPP	RBF
17	2008	Lee et al. [97]	ICA				✓										NE, TEP	RBF
18	2008	Cui et al. [98]	FDA								✓						NE, TEP	RBF, POLY
19	2008	Cui et al. [99]	SDA			✓											TEP	POLY
20	2008	Zhang and Qin [100]	ICA				✓				✓						TEP, WWTP, PenSim	RBF
21	2008	Lu and Wang [101]	PLS		✓	✓										✓	TEP	-
22	2008	He et al. [102]	FDA			✓				✓							TEP	RBF
23	2008	Cho [103]	FDA			✓											TEP	POLY
24	2008	Li and Cui [104]	SDA			✓					✓						TEP	POLY
25	2009	Li and Cui [105]	FDA			✓				✓	✓						TEP, PenSim	POLY, COS
26	2009	Zhang [106]	ICA			✓	✓				✓						TEP	RBF
27	2009	Zhang and Zhang [107]	ICA, PLS				✓		✓								TEP, PenSim	RBF
28	2009	Shao et al. [108]	PCA							✓		✓					NE, TEP	RBF
29	2009	Shao and Rong [109]	MVU			✓						✓					TEP	Manifold
30	2009	Shao et al. [110]	LPP			✓						✓					NE, TEP	Manifold
31	2009	Tian et al. [73]	ICA	✓			✓				✓						PenSim	RBF, POLY
32	2009	Liu et al. [111]	PCA					✓			✓		✓				NE, <b>BDP</b>	RBF
33	2009	Ge et al. [112]	PCA				✓	✓									NE, TEP	RBF

Table 3. Cont.

Year	Reference	Kernelized Method/s	Issues Addressed												Case Studies	Kernel/s Used
			A	B	C	D	E	F	G	H	I	J	K	L		
34	2009	Zhao et al. [113]					✓								NE, TEP	RBF
35	2009	Zhao et al. [114]	✓	✓		✓									TTP, PenSim	RBF
36	2010	Jia et al. [115]	✓	✓											NE, PenSim	RBF
37	2010	Cheng et al. [116]		✓			✓					✓			NE, TEP	RBF
38	2010	Alcala and Qin [117]				✓									CSTR	RBF
39	2010	Zhu and Song [118]				✓									TEP	RBF
40	2010	Zhang et al. [119]				✓							✓		CAP	RBF
41	2010	Zhang et al. [120]	✓	✓						✓					NE, PenSim	RBF
42	2010	Xu and Hu [121]				✓			✓						TEP	RBF
43	2010	Ge and Song [122]						✓							TEP	RBF
44	2010	Wang and Shi [123]					✓								WWTP, TEP	RBF
45	2010	Sumana et al. [124]		✓	✓										NE, TEP	RBF
46	2011	Sumana et al. [125]								✓					TEP	RBF
47	2011	Khediri et al. [126]										✓			NE, TEP	RBF
48	2011	Zhang and Ma [127]				✓									CAP, EFMF	RBF
49	2011	Zhang and Hu [128]	✓						✓						CAP, PenSim	RBF
50	2011	Zhang and Hu [129]				✓			✓					✓	NE, PenSim, EFMF	RBF
51	2011	Zhu and Song [130]				✓									TEP	RBF
52	2011	Yu [75]	✓	✓	✓										PenSim	RBF
53	2012	Khediri et al. [131]		✓											NE, SEP	RBF
54	2012	Rashid and Yu [82]	✓	✓		✓	✓								PenSim	RBF
55	2012	Zhang et al. [132]		✓									✓		CAP, PenSim	RBF
56	2012	Zhang and Ma [133]				✓	✓							✓	CAP	RBF
57	2012	Zhang et al. [134]		✓		✓									NE, TEP, EFMF	RBF
58	2012	Zhang et al. [135]	✓	✓					✓				✓		PenSim	-
59	2012	Yu [136]		✓	✓	✓									WWTP	RBF
60	2012	Guo et al. [137]		✓						✓					TEP	WAV
61	2012	Jia et al. [84]								✓					NE, PenSim	RBF, POLY, SIG
62	2012	Sumana et al. [138]		✓											TEP	POLY
63	2012	Wang et al. [139]	✓	✓											PenSim	POLY
64	2013	Liu et al. [140]				✓	✓	✓							CLG	RBF
65	2013	Peng et al. [141]							✓						NE, TEP, HSMP	RBF
66	2013	Peng et al. [79]				✓			✓						HSMP	RBF

Table 3. Cont.

Year	Reference	Kernelized Method/s	Issues Addressed												Case Studies	Kernel/s Used
			A	B	C	D	E	F	G	H	I	J	K	L		
67	2013	Wang et al. [142]	PCA	✓	✓			✓				✓			PenSim	POLY
68	2013	Jiang and Yan [143]	PCA					✓						✓	NE, CSTR, TEP	RBF
69	2013	Jiang and Yan [144]	PCA					✓							NE, TEP	RBF
70	2013	Zhang et al. [145]	ICA		✓		✓								<b>CAP</b>	RBF
71	2013	Zhang et al. [146]	PLS			✓			✓						NE, <b>EFMF</b>	RBF
72	2013	Zhang et al. [147]	PCA	✓	✓										PenSim, <b>EFMF</b>	-
73	2013	Zhang et al. [76]	VCA	✓	✓										<b>EFMF</b>	RBF
74	2013	Deng and Tian [148]	PCA			✓	✓	✓							NE, TEP	RBF
75	2013	Deng and Tian [149]	LPP									✓	✓		CSTR	RBF
76	2013	Deng et al. [150]	PCA			✓							✓		TEP	RBF
77	2013	Rong et al. [151]	LPP, FDA			✓	✓						✓		TEP, <b>WWTP</b>	RBF
78	2013	Hu et al. [152]	PLS	✓	✓										<b>PP</b> , PenSim	RBF
79	2013	Hu et al. [153]	PLS				✓				✓				NE, TEP	RBF
80	2014	Fan and Wang [66]	ICA		✓	✓	✓				✓				TEP	RBF
81	2014	Fan et al. [154]	ICA			✓	✓	✓			✓				NE, TEP	RBF
82	2014	Zhang et al. [155]	ICA			✓	✓								<b>EFMF</b>	-
83	2014	Zhang and Li [156]	PCA		✓	✓									<b>EFMF</b>	RBF
84	2014	Cai et al. [157]	ICA			✓	✓								NE, TEP	RBF
85	2014	Wang and Shi [158]	PLS				✓	✓							TEP	-
86	2014	Elshenawy and Mohamed [159]	PCA									✓			TEP	RBF
87	2014	Mori and Yu [160]	PCA, ICA, PLS	✓			✓		✓						PenSim	RBF
88	2014	Castillo et al. [161]	PCA			✓									<b>Air Heater</b>	RBF
89	2014	Vitale et al. [69]	PCA, PLS, FDA	✓		✓									NE, <b>PP</b> , <b>DP</b>	RBF, POLY
90	2014	Peng et al. [162]	PCA			✓					✓				CSTR	RBF
91	2014	Zhao and Xue [163]	T-PLS			✓			✓	✓					TEP	RBF+POLY
92	2014	Godoy et al. [164]	PLS			✓			✓	✓					NE	RBF
93	2014	Kallas et al. [165]	PCA			✓									NE, CSTR	RBF
94	2015	Ciabattoni et al. [166]	CVA		✓										<b>Microgrid</b>	RBF
95	2015	Vitale et al. [81]	PCA	✓		✓									NE, <b>DP</b> , <b>RCP</b>	RBF, POLY
96	2015	Li and Yang [167]	PCA							✓				✓	NE, TEP	RBF
97	2015	Liu and Zhang [168]	PLS		✓				✓						NE, PenSim	RBF
98	2015	Md Nor et al. [169]	FDA		✓	✓									TEP	-
99	2015	Yao and Wang [170]	PCA	✓						✓					PenSim	RBF

Table 3. Cont.

Year	Reference	Kernelized Method/s	Issues Addressed												Case Studies	Kernel/s Used	
			A	B	C	D	E	F	G	H	I	J	K	L			
100	2015	Wang and Yao [171]	PCA	✓				✓								NE, <b>SEP</b>	RBF
101	2015	Huang et al. [172]	CVA		✓					✓						TEP	RBF
102	2015	Zhang et al. [173]	PLS							✓						NE, <b>EFMF</b>	RBF
103	2015	Zhang et al. [174]	SFA		✓											NE, TEP	RBF
104	2015	Zhang et al. [175]	SFA, FDA				✓									CSTR	RBF
105	2015	Zhang et al. [176]	C-PLS				✓									PenSim	-
106	2015	Samuel and Cao [177]	CVA		✓											TEP	RBF
107	2015	Samuel and Cao [178]	CVA		✓											TEP	RBF
108	2015	Chakour et al. [179]	PCA		✓	✓								✓		TEP	RBF
109	2015	Jiang and Yan [180]	PCA												✓	NE, TEP	RBF
110	2015	Cai et al. [181]	CCA		✓										✓	NE, TEP	RBF
111	2015	Luo et al. [182]	GLPP							✓		✓				NE, TEP	RBF, HK
112	2015	Tang et al. [77]	VCA	✓	✓						✓					PenSim	RBF
113	2015	Bernal de Lazaro et al. [183]	PCA, FDA			✓				✓						TEP	RBF
114	2016	Bernal de Lazaro et al. [184]	PCA, ICA		✓			✓		✓						TEP	RBF
115	2016	Ji et al. [185]	PCA							✓						NE	RBF
116	2016	Xu et al. [186]	PCA		✓	✓		✓								NE, TEP	-
117	2016	Luo et al. [187]	GLPP								✓					NE, TEP	RBF
118	2016	Zhang et al. [188]	ICA		✓		✓									TEP	-
119	2016	Taouali et al. [189]	PCA								✓					CSTR	RBF
120	2016	Fazai et al. [190]	PCA											✓		CSTR, TEP	RBF
121	2016	Jaffel et al. [191]	PCA								✓			✓		TEP	RBF
122	2016	Mansouri et al. [192]	PCA					✓								NE, CSTR	-
123	2016	Botre et al. [193]	PLS					✓								CSTR	-
124	2016	Samuel and Cao [194]	PCA			✓	✓									TEP	RBF
125	2016	Ge et al. [195]	FDA			✓										<b>CSTH</b> , TEP	RBF
126	2016	Jia et al. [196]	PLS			✓	✓		✓							NE, <b>HGPWLTP</b>	RBF
127	2016	Jia and Zhang [197]	PLS		✓				✓							NE, TEP	RBF
128	2016	Jiang et al. [198]	PCA				✓								✓	TEP, CSTR	RBF
129	2016	Peng et al. [199]	PLS, Fuzzy C-means	✓	✓	✓			✓							<b>HSMP</b>	RBF
130	2016	Xie et al. [200]	PCA							✓	✓					NE, <b>BDP</b>	RBF
131	2016	Wang et al. [201]	PCR						✓							NE	RBF
132	2016	Huang and Yan [202]	PCA					✓								NE, TEP	RBF

Table 3. Cont.

Year	Reference	Kernelized Method/s	Issues Addressed												Case Studies	Kernel/s Used	
			A	B	C	D	E	F	G	H	I	J	K	L			
133	2016	Xiao and Zhang [203]	PCA, ICA			✓	✓									TEP	RBF
134	2016	Feng et al. [204]	FDA			✓	✓									TEP	RBF
135	2016	Sheng et al. [205]	C-PLS							✓						NE, TEP	RBF
136	2016	Zhang et al. [206]	PLS, PCA			✓				✓					✓	<b>CAP</b>	RBF
137	2017	Jaffel et al. [207]	PCA								✓				✓	CSTR, TEP	RBF
138	2017	Lahdhiri et al. [208]	PCA								✓				✓	NE, CSTR, <b>AIRLOR</b>	RBF
139	2017	Lahdhiri et al. [209]	PCA								✓				✓	NE, CSTR	RBF
140	2017	Mansouri et al. [210]	PLS		✓					✓						CSEC, <b>GCND</b>	RBF
141	2017	Mansouri et al. [211]	PCA							✓						CSEC	-
142	2017	Sheriff et al. [212]	PCA		✓											CSTR	RBF
143	2017	Cai et al. [213]	ICA			✓	✓									NE, TEP	RBF
144	2017	Zhang et al. [214]	ECA		✓	✓		✓								TEP	RBF
145	2017	Zhang et al. [215]	SFA	✓	✓										✓	NE, PenSim	RBF
146	2017	Zhang and Tian [216]	SFA	✓	✓											PenSim	POLY
147	2017	Zhang et al. [217]	PCA			✓										<b>EFMF</b>	-
148	2017	Zhang et al. [218]	PCA, LLE												✓	<b>EFMF</b>	-
149	2017	Zhang et al. [219]	PCA		✓											NE, <b>SEP</b>	RBF
150	2017	Deng et al. [220]	PCA						✓							TEP	RBF
151	2017	Deng et al. [221]	PCA		✓				✓							NE, CSTR	RBF
152	2017	Deng et al. [222]	PCA, FDA			✓									✓	NE, CSTR	RBF
153	2017	Tan et al. [223]	CVA		✓											<b>MFF</b>	-
154	2017	Shang et al. [224]	CVA		✓											CSTR	RBF
155	2017	Li et al. [225]	DLV		✓											<b>HSMP</b>	RBF
156	2017	Wang and Jiao [226]	LS							✓						NE, TEP	RBF
157	2017	Wang et al. [227]	DD						✓							NE, TEP	RBF
158	2017	Wang et al. [228]	EDA	✓		✓										PenSim	RBF
159	2017	Jiao et al. [229]	PLS							✓						NE, TEP	RBF
160	2017	Huang and Yan [230]	PCA							✓						NE, TEP	RBF
161	2017	Yi et al. [231]	PLS			✓			✓							TEP, <b>AEP</b>	-
162	2017	Md Nor et al. [232]	FDA		✓	✓										TEP	RBF
163	2017	Du et al. [233]	ICA				✓									<b>EFMF</b>	-
164	2017	Zhang and Zhao [234]	PCA, Fuzzy C-means		✓										✓	TEP, <b>MFF</b>	RBF
165	2017	Zhou et al. [235]	RPLVR				✓		✓							NE, TEP	-



Table 3. Cont.

Year	Reference	Kernelized Method/s	Issues Addressed												Case Studies	Kernel/s Used
			A	B	C	D	E	F	G	H	I	J	K	L		
166	2017	Gharahbagheri et al. [236]	PCA			✓									DTS, FCCU, TEP	RBF
167	2017	Gharahbagheri et al. [237]	PCA			✓									NE, FCCU, TEP	RBF
168	2017	Fu et al. [68]	PCA, PLS								✓				NE, <b>GMP, BDP, Mixing</b>	RBF
169	2017	Galiaskarov et al. [238]	FDA			✓							✓	<b>Pyrolysis gas furnace</b>	POLY	
170	2017	Zhu et al. [239]	ICA			✓	✓							TEP	RBF, POLY, SIG	
171	2017	Zhu et al. [240]	CCA							✓				TEP	RBF	
172	2018	Liu et al. [241]	CCA		✓	✓				✓				<b>CAP</b>	RBF	
173	2018	Wang and Jiao [242]	PLS							✓				NE, TEP	RBF	
174	2018	Wang [243]	PLS					✓	✓					NE, CSTR	RBF	
175	2018	Huang and Yan [244]	PCA							✓				NE, TEP	RBF	
176	2018	Huang and Yan [245]	PCA					✓	✓					NE, TEP, <b>IPOP</b>	RBF	
177	2018	Fezai et al. [246]	PCA							✓			✓	NE, TEP	RBF	
178	2018	Fezai et al. [247]	PCA			✓				✓	✓		✓	<b>AIRLOR</b>	RBF	
179	2018	Mansouri et al. [248]	PCA		✓			✓						NE, CSEC	RBF	
180	2018	Jaffel et al. [249]	PCA			✓				✓			✓	CSTR	RBF	
181	2018	Lahdhiri et al. [250]	PCA							✓	✓		✓	NE, TEP	RBF	
182	2018	Tan and Cao [251]	PCA			✓								NE, TEP	RBF	
183	2018	He et al. [252]	LPP	✓				✓			✓			PenSim, <b>HSMP</b>	RBF	
184	2018	Navi et al. [253]	PCA		✓	✓							✓	IGT	RBF	
185	2018	Chakour et al. [254]	PCA			✓								TEP, <b>Weather station</b>	RBF	
186	2018	Deng and Wang [255]	PCA				✓							NE, TEP	RBF	
187	2018	Deng et al. [256]	PCA					✓			✓			NE, TEP	RBF, POLY	
188	2018	Deng et al. [257]	PCA			✓								NE, TEP	RBF	
189	2018	Deng et al. [258]	FDA			✓		✓			✓			TEP	RBF	
190	2018	Zhang et al. [259]	SFA	✓	✓	✓								NE, CSTR	RBF	
191	2018	Shang et al. [260]	AMD		✓									NE, TEP	POLY	
192	2018	Jiang and Yan [261]	PCA										✓	NE, CSTR	-	
193	2018	Feng et al. [262]	ICA				✓	✓						<b>EFMF</b>	RBF	
194	2018	Zhao and Huang [263]	PCA, DISSIM					✓						<b>TPP, CPP</b>	RBF	
195	2018	Zhai et al. [264]	NNMF			✓								PenSim	-	
196	2018	Ma et al. [265]	ICA			✓	✓							TEP	RBF	
197	2018	Lu et al. [266]	CVA, LPP, FDA	✓	✓						✓			TEP	HK	
198	2018	Li et al. [267]	PCA										✓	NE, <b>CPP</b>	-	

Table 3. Cont.

Year	Reference	Kernelized Method/s	Issues Addressed												Case Studies	Kernel/s Used	
			A	B	C	D	E	F	G	H	I	J	K	L			
199	2018	Chu et al. [268]	PLS			✓			✓							<b>DMCPP</b>	RBF
200	2019	Zhai and Jia [269]	NNMF			✓										NE, PenSim	RBF
201	2019	Fezai et al. [270]	PCA					✓		✓		✓				PV	RBF
202	2019	Fazai et al. [271]	PLS					✓		✓		✓				TEP	RBF
203	2019	Deng and Deng [272]	PCA		✓			✓								NE, TEP	RBF
204	2019	Cui et al. [273]	PCA									✓			✓	NE, TEP	RBF, Manifold
205	2019	Pilario et al. [67]	CVA		✓		✓	✓		✓						NE, CSTR	RBF+POLY
206	2019	Lahdhiri et al. [274]	PCA			✓				✓	✓		✓			<b>AIRLOR</b>	RBF
207	2019	Liu et al. [275]	ICA			✓	✓		✓							<b>GHP</b>	RBF
208	2019	Liu et al. [276]	ICA			✓	✓			✓						TEP	RBF
209	2019	Yu et al. [277]	CCA			✓			✓		✓					NE, TEP	RBF
210	2019	Guo et al. [278]	PCA			✓					✓					NE, TEP	RBF
211	2019	Wu et al. [279]	PCA			✓				✓	✓					NE, TEP	RBF
212	2019	Harkat et al. [280]	PCA							✓						NE, TEP	RBF
213	2019	Ma et al. [281]	CVA, EDA		✓	✓			✓				✓			<b>HSMP</b>	-
214	2019	Zhang et al. [282]	ELM									✓				NE, CSTR	RBF
215	2019	Peng et al. [83]	ECA	✓	✓	✓										NE, PenSim	RBF
216	2019	Peng et al. [283]	ICA, EDA		✓	✓	✓				✓		✓	✓		TEP	-
217	2019	Yan et al. [284]	PCA, PLS						✓					✓		NE, TEP	RBF
218	2019	Huang et al. [285]	DL			✓										NE, <b>CSTH, AEP</b>	RBF
219	2019	Li and Zhao [80]	DFFDA			✓							✓	✓		NE, <b>IMP, CFPP</b>	RBF
220	2019	Zhou et al. [286]	PCA								✓					NE, TEP	RBF
221	2019	Deng et al. [287]	PCA		✓									✓		TEP	RBF
222	2019	Wang et al. [288]	PCA			✓				✓	✓					CSTR, <b>HSMP</b>	RBF
223	2019	Zhu et al. [289]	PLS			✓				✓	✓					TEP	RBF
224	2019	Xiao [290]	CVA, LPP		✓							✓				TEP	HK
225	2019	Xiao [291]	CVA		✓			✓								TEP	RBF
226	2019	Shang et al. [292]	PCA										✓			TEP	RBF
227	2019	Geng et al. [293]	PCA			✓							✓			TEP	RBF
228	2019	Md Nor et al. [294]	FDA		✓	✓										TEP	-
229	2019	Tan et al. [295]	PCA		✓					✓						NE, <b>MFF</b>	NSDC
230	2019	Tan et al. [296]	PCA		✓					✓						NE, <b>MFF</b>	NSDC

## 4. Review Findings

In this section, the major issues on kernel-based process monitoring, as identified and presented in Table 2, are discussed one by one. We first motivate why they are important and then give examples of how they were addressed by many researchers through the years.

### 4.1. Batch Process Monitoring

Monitoring batch processes is important so as to reduce batch-to-batch variability and maintain the quality of products [70]. The first application of kernel PCA to process monitoring was in a continuous process [24], wherein the plant data set is a matrix of  $M$  variables  $\times N$  samples (2-D) (see Section 2). In contrast, for a batch process, the plant data set is a tensor of  $K$  batches  $\times M$  variables  $\times N$  samples (3-D) and, hence, must be handled differently. A multi-way approach is commonly adopted, where tensor data is unfolded into matrix data either variable-wise or batch-wise so that the kernel MSPM method can now apply. This led to multi-way kernel PCA [71], multi-way kernel ICA [72,73], multi-way kernel FDA [74,75], and so on. Variable correlation analysis (VCA) and its kernelized version was also proposed for batch process monitoring in [76,77]. Common batch process case studies include the fed-batch fermentation process for producing penicillin (PenSim) available as a simulation package from Birol et al. [78], the hot strip mill process (HSMP) as detailed in [79], the injection moulding process (IMP) [80], and other pharmaceutical processes [69,81].

If batch data sets have uneven lengths, the trajectories must be synchronized prior to analysis. Dynamic time warping (DTW) is one such technique to handle this issue, as adopted by Yu [75] and Rashid and Yu [82]. Another problem is related to the multi-phase characteristic of batch process data. Since a whole batch consists of steady-state and transition phases, then each phase must be modelled differently. Phase division has been employed to address this issue, as did Tang et al. [77] and Peng et al. [83]. In all these studies, the RBF and POLY kernels were mostly used to generate nonlinear features for process monitoring. But in particular, Jia et al. [84] has found that the POLY kernel is optimal for the PenSim case study, as calculated by a genetic algorithm (GA).

We refer the reader to the reviews by Yao and Gao [297] and Rendall et al. [298] for more information on batch process data analytics beyond the application of kernel methods.

### 4.2. Dynamics, Multi-Scale, and Multi-Mode Monitoring

Recall that in the framework of Figure 2, a column vector of samples at instant  $k$  is used to generate the statistical index for that instant. This scheme is merely static, however. It does not account for the trends and dynamic behaviors of the plant in the statistical model. Dynamic behaviors manifest in the data as serial correlations or trends at multiple time scales, which can arise from varying operating conditions. It is important to address both nonlinear and dynamic issues, as they can improve the accuracy of fault detection significantly [25].

To address dynamics, features must be extracted from time-windows of data samples at once (lagged samples) rather than sample vectors at one instant only. Dynamic extensions of kernel PCA [85,96,115,116,260], kernel PLS [101], and kernel ICA [66] have used this approach. In addition, some MSPM tools are inherently capable of extracting dynamic features effectively, such as canonical variate analysis (CVA) [299], slow feature analysis (SFA) [300], and dynamic latent variable models (DLV). Kernel CVA is the kernelized version of CVA and is used in many works [67,166,172,177,178,223,224,281,290,291]. Meanwhile, kernel slow feature analysis has appeared in [174,215,216,259], and more recently, the kernel dynamic latent variable model was proposed in [225]. The details of kernel CVA, kernel SFA, and kernel DLV can be found in these references. For mining the trends in the data at multiple time scales, wavelet analysis is commonly used. Multi-scale kernel PCA was first proposed by Deng and Tian [91], followed by similar works in [94,95,134,169,210], which includes multi-scale kernel PLS and multi-scale kernel FDA. A wavelet kernel was also proposed by Guo et al. [137], which was applied to the Tennessee Eastman Plant (TEP).

Multi-modality is a related issue found in processes that are designed to work at multiple operating points [38]. Figure 7 shows an example of a data set taken from the multiphase flow facility at Cranfield University [18], which exhibits multi-modality on the air flow measurements. The challenge is having to distinguish if transitions in the data are due to a change in operating mode or due to a fault. If this issue is not addressed, the changes in operating mode will trigger false alarms [38]. To address this issue, Yu [75] used  $k$ -nearest neighbors to classify the data prior to performing localized kernel FDA for batch process monitoring. Meanwhile, Khediri et al. [131] used kernel K-means clustering to identify the modes, and then support vector data description (SVDD) to detect faults in each cluster. Other ways to identify modes include a kernel Gaussian mixture model [136], hierarchical clustering [139,142], and kernel fuzzy C-means [199,234]. More recently, Tan et al. [295,296] proposed a new kernel design, called non-stationary discrete convolution kernel (NSDC), for multi-mode monitoring (see Section 4.7). The NSDC kernel was found to yield better detection performance than the RBF kernel based on the multiphase flow facility data [18].

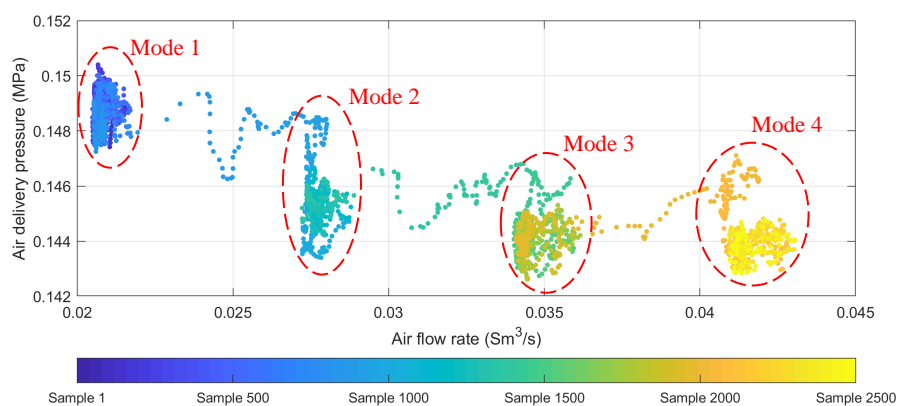


Figure 7. Illustration of multi-modality in process operations.

#### 4.3. Fault Diagnosis in the Kernel Feature Space

Diagnosis is a key process monitoring task. When a fault is detected in the plant, it is imperative to determine where did it occur, what type of fault it is, and how large its magnitude. The actual issue is that when nonlinear feature extraction is employed, fault diagnosis is more difficult to perform.

##### 4.3.1. Diagnosis by Fault Identification

The usual practice is to first identify the faulty variables based on their influence to the value of the statistical index. This scheme is called *fault identification*. It is beneficial to identify which variables are associated to the fault, especially when the plant is highly integrated and the number of process variables is large [1]. There are 2 major ways to perform fault identification: variable contributions and variable reconstructions. Variable contributions are computed by taking the first-order Taylor series expansion of the statistical index to reveal which variables contribute the most to its value [87]. In the other approach, each variable is reconstructed in terms of the remaining variables to estimate the fault magnitude (the amount of reconstruction) along that direction [117]. Hence, variables with the largest amount of reconstructions are associated to the fault. Results can be visualized in contribution plots or contribution maps [301] to convey the diagnosis.

Fault identification is straightforward if the feature extraction involves only a linear machine. For kernel methods, however, it is complicated by the fact that the data went through a nonlinear mapping. This is because both approaches entail differentiating the statistical index, which is difficult if the chain involves a kernel function [86]. Nevertheless, many researchers have derived analytical expressions for either kernel contributions-based diagnosis [66,79,81,83,87,94,119,127,133,136,146,150,156,157,162,164,194,213,241,268,275,276,278,279,288,289,293] or kernel reconstructions-based diagnosis [86,117,140,155,161,163,176,217,236,254,265,285]. However, most derivations are applicable

only when the kernel function is the RBF, Equation (5). In one approach, Tan and Cao [251] proposed a new deviation contribution plot to perform fault identification for any nonlinear feature extractor.

#### 4.3.2. Diagnosis by Fault Classification

The fault identification approach assumes that no prior fault information is available for making a diagnosis. If fault information is available, then the learning problem becomes that of finding the boundary between normal and faulty samples or the boundary between different fault types, within the feature space (see Section 2.2). This learning problem pertains to *fault classification*, and the three common approaches are similarity factors, discriminant analysis, and SVMs.

The similarity factor method (SFM) was proposed by Krzanowski [302] to measure the similarity of two data sets using PCA. For fault classification, the idea is to compute the similarity between the test samples against a historical database of fault samples, and find the fault type that is most similar. A series of works by Deng and Tian [91,95,148] used SFM for diagnosis, after performing multi-scale KPCA for fault detection. Ge and Song [303] also proposed the ICA similarity factor, although it was not performed in a kernel feature space. SFM was also applied to features derived from kernel slow feature analysis (SFA) [175] and serial PCA [257].

Discriminant analysis, notably Fisher discriminant analysis (FDA), is a linear MSPM method that transforms the data as in Equation (1) where the weights are obtained by maximizing the separation of samples from different classes while minimizing the scatter within each class [1]. This means that the generated features from FDA are discriminative in nature. Kernel FDA, its nonlinear extension, is used extensively such as in [74,75,80,92,98,102,103,105,118,130,151,169,175,183,195,204,222,232,238,258,266,294]. One variant of FDA is exponential discriminant analysis (EDA) which solves the singularity problem in the FDA covariance matrices by taking their exponential forms [281,283]. Another variant is scatter-difference-based discriminant analysis (SDA), whose kernel version first appeared in [99], and then in [104,124]. SDA differs from FDA in that the difference of between-class scatter and within-class scatter matrices is maximized rather than their ratio, and hence avoids any matrix inversion or singularity problems [99]. Lastly, a kernel PLS discriminant analysis variant is used in batch process monitoring in [69].

SVM is a well-known method of choice for classification in machine learning, originally proposed by Cortes and Vapnik [304]. It is also regarded as the most popular kernel method, according to Domingos [50], although he also advocates that simpler classifiers (e.g., kNN) must be tried first before SVM [40]. In this regard, Zhang [106,305] used SVM on kernel PCA and kernel ICA features to perform diagnosis. Xu and Hu [121] and Xiao and Zhang [203] used a similar approach for classification, but also employs multiple kernel learning [306]. Meanwhile, Md Nor et al. [232] used SVM on the features from multi-scale kernel FDA. Aside from SFM, FDA, and SVM, an ANN-based fault classifier was also used by Bernal de Lazaro [183] on kernel PCA and kernel FDA features.

The Tennessee Eastman Plant (TEP) is usually the case study in most of these papers, as it contains samples at normal plant operation as well as from each of 20 different fault scenarios. Once the fault classifier is trained, it can automatically assign every new test sample as to normal or to any fault scenario it was trained on. However, the fault classification methods require a database of samples from many different fault scenarios *a priori*, in order to provide a comprehensive diagnosis.

#### 4.3.3. Diagnosis by Causality Analysis

So far, the above methods are unable to perform a root cause diagnosis. Root cause diagnosis is valuable for cases when the fault has already propagated to multiple locations, making it difficult to locate its origin. To perform such a task, the causal relationships between process variables must be known so that the fault propagation can be traced throughout the plant [307]. Causal information can be supplied by process knowledge, plant operator experience, or model-based principles. One such work is by Lu and Wang [101], who used a signed digraph (SDG) model of the TEP consisting of 127 nodes and 15 root-cause nodes, and then used 20 local dynamic kernel PLS models for the

subsystems. However, as a consequence of the kernel mapping, traversing the SDG backwards is difficult since it is impossible to find the inverse function from the kernel feature space to the original space [101]. Hence, the diagnosis was only performed qualitatively in that work [101].

The Bayesian network is an architecture for causality analysis, where the concepts of Granger causality and transfer entropy are used to define if one variable is caused by another based on their time series data. In 2017, Gharahbagheri et al. [236,237] used these concepts together with the residuals from kernel PCA models to generate a causal map for a fluid catalytic cracking unit (FCCU) and the TEP. A statistical software called *Eviews* was used to perform causality analysis.

In the future, fault diagnosis by causality analysis can potentially benefit from the combination of knowledge-, physics-, and data-driven approaches [1].

#### 4.4. Handling Non-Gaussian Noise and Outliers

Recall that in the feature extraction step in Figure 2, it is desired to yield features that are mutually independent so that the  $T^2$  statistical index can be built. However, previous methods such as PCA and PLS (even their kernelized versions) may fail to yield such features, especially if the data is laden with non-Gaussian noise or outliers. This issue is widely recognized in practice [25]. Instinctively, MSPM methods can be used for detecting outliers. However, if outliers are present in the training data itself, the accuracy of MSPM algorithms will be seriously affected.

Independent components analysis (ICA) and its kernelized version, kernel ICA, are widely used MSPM methods that can handle the non-Gaussianity issue. Here, the data is treated as a mixture of independent source signals, so that the aim of ICA is to de-mix the data and recover these sources [308]. To do this, the projection matrix in ICA,  $\mathbf{W}_n$  (also known as a de-mixing matrix), is chosen so that the ICA features are as statistically independent as possible [308]. More concretely, the goal is usually to maximize *negentropy*, which is a measure of the distance of a distribution from Gaussianity [309]. Kernel ICA can be performed by doing kernel PCA for whitening, followed by linear ICA, as did many researchers [66,72,73,82,90,97,100,106,107,133,140,145,154,155,157,188,203,213,233,239,265,275,276,283,305]. A variant of kernel ICA that avoids the usual KPCA-ICA combination is also proposed by Feng et al. [262]. Aside from kernel ICA, the non-Gaussianity issue can also be handled using a kernel Gaussian mixture model [136], the use of statistical local approach for building the statistical index [112], and kernel density estimation (KDE) for threshold setting [67,194,251].

To handle outliers in the data, Zhang et al. [134] and Deng and Wang [255] incorporated a sliding median filter and a local outlier factor method, respectively, into kernel PCA. Other outlier-robust methods include the spherical kernel PLS [153], the joint kernel FDA [204] and the kernel probabilistic latent variable regression model [235].

#### 4.5. Improved Sensitivity and Incipient Fault Detection

Despite the use of advanced MSPM tools, it may be desired to improve their detection sensitivity further. This is beneficial in particular for detecting incipient faults, which are small-magnitude faults with a drifting behavior. These faults are difficult to detect at the initial stage because they are masked by noise and process control [67]. Yet because they are drifting, they can seriously escalate if no action takes place. Kernel MSPM solutions to these issues already exist, which we review as follows.

An early approach for improved detection is dissimilarity analysis (DISSIM), proposed by Kano et al. [310]. DISSIM is mathematically equivalent to PCA but its statistical index is different from the  $T^2$  in that it quantifies the dissimilarity between data distributions. Its kernel version, kernel DISSIM, was developed by Zhao et al. [113], and further used in Zhao and Huang [263]. The concept of dissimilarity was also adopted by Pilario et al. [67] and Xiao [291] for kernel CVA and Rashid and Yu [311] for kernel ICA. Related to DISSIM is statistical pattern analysis (SPA), used in [148,221,258] for kernel PCA. The idea of SPA, as proposed by He and Wang [312], is to build a statistical index from the dissimilarity between the higher-order statistics of two data sets.

Another approach is to use an exponentially weighted moving average filter (EWMA) to increase the sensitivity for drifting faults, as did Yoo and Lee [88], Cheng et al. [116], Fan et al. [154], and Peng et al. [283]. The shadow variables by Feng et al. [262] also involve applying EWMA on the statistical indices for smoothing purposes as well. For batch processes, a method for detecting weak faults is also proposed by Wang et al. [139]. The works of Jiang and Yan [143,144] improved the sensitivity of kernel PCA by investigating the rate of change of the statistical index and by giving a weight to each feature. Lastly, a new statistic based on the generalized likelihood ratio test (GLRT) can also improve detection for kernel PCA and kernel PLS, as shown by Mansouri et al. [192,193,210,270,271].

#### 4.6. Quality-Relevant Monitoring

Before the widespread use of MSPM methods, the traditional approach to process monitoring is to monitor only the *quality variables* [8] as embodied by statistical quality control. MSPM methods are more beneficial in that it utilizes the entire plant data set rather than just the quality variables to perform fault detection. However, as noted by Qin [25], it is imperative to link the results from MSPM methods to the quality variables. The kernel MSPM methods discussed thus far have not yet established this link. This issue can be addressed by performing quality-relevant monitoring.

Partial least squares (PLS) is an MSPM method associated with quality-relevant monitoring, as it finds a relationship between the process and quality variables. The first kernel PLS application was in a biological anaerobic filter process (BAFP) by Lee et al. [89], where the quality variables are the total oxygen demand of the effluent and flow rate of exiting methane gas. Zhang and Zhang [107] combined ICA and kernel PLS for monitoring the well-known penicillin fermentation (PenSim) process and predicting the CO<sub>2</sub> and dissolved O<sub>2</sub> concentrations. Hierarchical kernel PLS, dynamic hierarchical kernel PLS, and multi-scale kernel PLS were introduced in [128,135], and [129], respectively. Total PLS (T-PLS) was proposed to make PLS more comprehensive, and its kernel version was developed by Peng et al. [79,141]. The application was in the HSMP, wherein both quality-related and non-quality-related faults were investigated. Further developments on kernel PLS can be found in [146,160,163,164,168,173,196,197,199,206,229,231,242,243,268,284]. Concurrent PLS was also proposed to solve some drawbacks of the T-PLS. Kernel concurrent PLS was developed by Zhang et al. [176] and Sheng et al. [205].

The other more recent MSPM tool for relating process and quality variables is canonical correlation analysis (CCA). CCA is different from PLS in that it finds projections that maximize the correlation between two data sets. Kernel CCA first appeared in process monitoring as a modified ICA by Wang and Shi [123], but it was not utilized for quality-relevant monitoring. The same is true in Cai et al. [181], where kernel CCA was merely used to build a complex network for the process. In 2017, Zhu et al. [240] first proposed the kernel concurrent CCA for quality-relevant monitoring. Liu et al. [241] followed with its dynamic version. In a very recent work by Yu et al. [277], a faster version of kernel CCA was proposed, to be discussed later in Section 4.8.

#### 4.7. Kernel Design and Kernel Parameter Selection

The issue of kernel design is often cited as the reason why researchers would prefer to use other nonlinear techniques over kernel methods. It is difficult to decide which kernel function to use (see Equations (5)–(7)) and how kernel parameters should be chosen. (Note, however, that decisions like these also exist in ANNs, e.g., how to set the depth of the network, number of hidden neurons, and learning rate, which activation function to use and which regularization method to use.) These choices also depend on the decisions made at other stages of process monitoring. For instance, choosing one kernel function over another may change the number of retained kernel principal components necessary for good performance. Moreover, the quality of the training data can influence all these decisions. Even if these parameters were carefully tuned based on fixed data sets for training and validation, the detection model may still yield too many false alarms if the data sets are not

representative of all behaviors of the normal process. Process monitoring performance greatly depends on these aspects. We review existing efforts that address these issues, as follows.

#### 4.7.1. Choice of Kernel Function

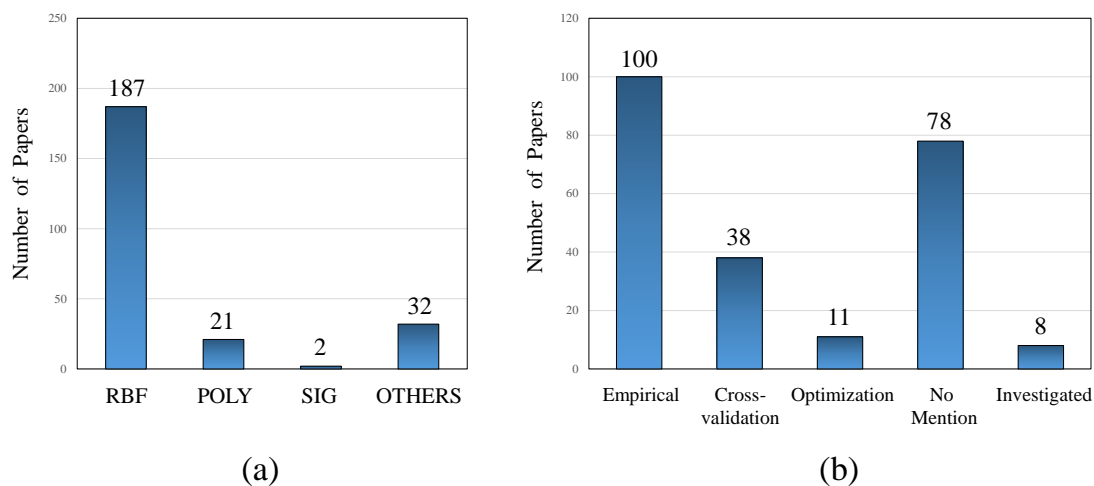
The main requirement for a kernel function to be valid is to satisfy Mercer's condition [22]. According to Mercer's theorem, as quoted from [313]: A necessary and sufficient condition for a symmetric function  $k(\cdot, \cdot)$  to be a kernel is that for any set of samples  $\mathbf{x}, \dots, \mathbf{x}_\ell$  and any set of real numbers  $\lambda_1, \dots, \lambda_\ell$ , the function  $k(\cdot, \cdot)$  must satisfy:

$$\sum_{i=1}^{\ell} \sum_{j=1}^{\ell} \lambda_i \lambda_j k(\mathbf{x}_i, \mathbf{x}_j) \geq 0 \quad (8)$$

which translates to the function  $k(\cdot, \cdot)$  being *positive definite*.

This means that if a function satisfies the condition in Equation (8), it can act as a dot product in the mapping of  $\mathbf{x}$  defined by  $\phi(\cdot)$ , and hence, it is a valid Mercer kernel function. If  $k(\cdot, \cdot)$  acts as a dot product, then for any two samples,  $\mathbf{x}$  and  $\mathbf{z}$ , the function is symmetric, i.e.,  $k(\mathbf{x}, \mathbf{z}) = k(\mathbf{z}, \mathbf{x})$ , and also satisfies the Cauchy-Schwarz inequality:  $k^2(\mathbf{x}, \mathbf{z}) \leq k(\mathbf{x}, \mathbf{x})k(\mathbf{z}, \mathbf{z})$  [313].

Although many kernel functions exist [44,314], only a few common ones are being used in process monitoring, namely, Equations (5)–(7). We identified the kernels used in each of the 230 papers included in this review. In the tally shown in Figure 8a, the RBF kernel is found to be the most popular choice, by a wide margin. Even outside the process monitoring community, the Gaussian RBF kernel (also known as the squared exponential kernel) is the most widely used kernel in the field of kernel machines [314], possibly owing to its smoothness and flexibility. Other kernels found from the review are the cosine kernel [105], wavelet kernel [137], the recent non-stationary discrete convolution kernel (NSDC) [295,296], and the heat kernel [182,266,290] for manifold learning (see Section 4.9).



**Figure 8.** (a) Number of papers that cited the use of which kernel functions; (b) Number of papers that cited the use of which kernel parameter selection routes. Note: Papers can appear in more than one column, hence, the numbers will not add to 230 (the total number of reviewed papers).

Other advances are related to the kernel design itself. For instance, Shao et al. [108] and Luo et al. [182] proposed data-dependent kernels for kernel PCA, which is used to learn manifolds. A robust alternative to kernel PLS is proposed by Hu et al. [153] which uses a sphered kernel matrix. Meanwhile, Zhao and Xue [163] used a mixed kernel for kernel T-PLS to discover both local and global patterns. The mixed kernel consists of a convex addition of the RBF and POLY kernels. Mixed kernels were also used by Pilario et al. [67] for kernel CVA, but motivated by monitoring incipient faults.



This additive principle was also used to design a kernel for batch processes by Yao and Wang [170]. More recently, Wang et al. [288,289] proposed to use the first-order expansion of the RBF kernel to save computational cost. However, it is not clear if the new design retains the same flexibility of the original RBF kernel to handle nonlinearity, or if it compares to polynomial kernels of the same order.

#### 4.7.2. Kernel Parameter Selection

The kernel parameters for the RBF, POLY, and SIG kernels in Equations (5)–(7) are the kernel bandwidth,  $c$ , the polynomial degree,  $d$ , and the sigmoid scale  $a$  and bias  $b$ . These kernels satisfy Mercer's conditions for  $c > 0$ ,  $d \in \mathbb{N}$ , and only some combinations of  $a$  and  $b$  [22,67]. There are currently no theoretical basis on how to specify the values of these parameters, yet they must be specified prior to performing any kernel method. We review some of the existing ways to obtain their values, as follows.

We have tallied the various parameter selection routes used by the 230 papers included in this review. Based on the results in Figure 8b, the most popular approach is to select them empirically. For the RBF kernel,  $c$  is usually computed based on the data variance ( $\sigma^2$ ) and dimensionality ( $m$ ), i.e.,  $c = r m \sigma^2$  [24,72,96,97], where  $r$  is an empirical constant. Another heuristic is based solely on the dimensionality, such as  $c = 5m$  [86–88] or  $c = 500m$  [66,118,130,204] for the TEP case study. For the TEP alone, many values were used, such as  $c = 6000$  [157,213],  $c = 1720$  [177],  $c = 4800$  [205],  $c = 3300$  [220], and so on. However, note that the appropriate value of  $c$  does not depend on the case study, but rather on the characteristics of the data that enters the kernel mapping. Hence, various choices will differ upon using different data pre-processing steps, even for the same case study. Other notable heuristics for  $c$  can be found in [68,126,131,164,248,280].

A smaller number of papers have used cross-validation to decide kernel parameter values. In this scheme, the detection model is tuned according to some objective, such as minimizing false alarms, using a *validation* data set that must be independent from the training data [67]. Another scheme is to perform  $k$ -fold cross-validation, as did [85], in which the data set is split into  $k$  groups:  $k - 1$  groups are used for training, while the remaining group is used for validation, and then repeat  $k$  times for different held-out data. Typically,  $k = 5$  or 10. Grid search is a common approach for the tuning stage, where the kernel parameters are chosen from a grid of candidates, as did [67,79,98,121,124,141,151,170,171,195,201,215,259]. Based on a recent study by Fu et al. [68], cross-validation was found to yield better estimates of the kernel parameters than the empirical approach.

A more detailed approach to compute kernel parameters is via optimization. It is known that if certain objectives are set, these parameters will have an optimal value. For instance, as explained by Bernal de Lazaro [184], if the RBF kernel bandwidth  $c$  is too large, the model loses the ability to discover nonlinear patterns, but if it is too small, the model will become too sensitive to the noise in the training data. Hence, the value of  $c$  can be searched such that the false alarm rate is minimum and the detection rate is maximum [184]. Exploring these trade-offs is key to the optimization procedure. Other criteria for optimizing kernel parameters were proposed in [183]. Some search techniques include the bisection method [162], Tabu search [247,250,274], particle swarm optimization [184,276], differential evolution [184], and genetic algorithm [84,93,102,108,154]. More recent studies have emphasized that kernel parameters must be optimized simultaneously with the choice of latent components (e.g., no. of kernel principal components) since these choices depend on each other [67,68].

Finally, there are also some papers that investigated the effect of varying the kernel parameters and presented their results (see [67,80,98,165,185,256,295,296]). In case the reader is interested in the investigation, we have provided a MATLAB code for visualizing the contours of kernel PCA statistical indices for any 2-D data set, available in [315]. This code was used to generate one of the figures in [67]. Understanding the effect of kernel parameters and the kernel function is important, especially as process monitoring methods become more sophisticated in the future.

#### 4.8. Fast Computation of Kernel Features

Recall in Section 2.3 that one of the issues of kernel methods is scalability. This is because the computational complexity of kernel methods grow in proportion to the size of training data. Hence, although they are fast to train, they are slow in making predictions [45]. Addressing the scalability of kernel methods is important, especially since samples are now being generated at large volumes in the plant [8]. The time complexity of naïve kernel PCA for the online testing phase is  $O(N^2)$ , where  $N$  is the number of training samples. Assuming that a typical CPU can do  $10^8$  operations in one second [316], kernel PCA can only allow at most  $10^4$  training samples if a prediction is desired within a second as well. In the following, we review the many approaches adopted by process monitoring researchers to compute kernel features faster.

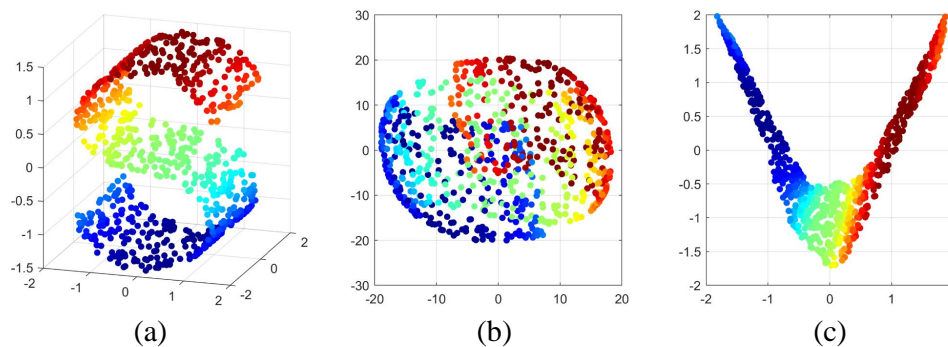
An early approach to reduce the computational cost of kernel MSPM methods is to select only a subset of the training samples so that their mapping is as expressive as if the entire data set was used. By reducing the number of samples, the kernel matrix reduces in size, and hence the transformation in Equation (4) can be computed faster. Feature vector selection (FVS) is one such method in this regard, as proposed by Baudat and Anouar [317], and then adopted by Cui et al. [98] for kernel PCA based process monitoring. FVS aims to preserve the geometric structure of the kernel feature space by an iterative error minimization process. Cui et al. [98] have shown that for the TEP, even if only 30 out of the 480 samples were selected by FVS and stored by the model, the average fault detection rate has changed only by 0.7%. FVS was further adopted in [77,104,105,125,149,256]. A related feature points extraction scheme by Wang et al. [142] was also proposed for batch processes. Another idea is *similarity analysis*, wherein a sample is rejected from the mapping if it is found to be similar to the current set by some criteria (This is not to be confused with the similarity factor method, SFM, discussed in Section 4.3.2). Similarity analysis was adopted by Zhang and Qin [100] and Zhang [106]. Meanwhile, Guo et al. [278] reformulated kernel PCA itself to sparsify the projection matrix using elastic net regression. Other techniques for sample subset selection includes feature sample extraction [73], the use of fuzzy C-means clustering [159], reduced KPCA [207], partial KPCA [249], and dictionary learning [246,250,270,271,274]. These methods are efficient enough to warrant an online adaptive implementation (see Section 4.10).

The other set of approaches involves a low-rank approximation of the kernel matrix for large-scale learning. Nyström approximation and random Fourier features are the typical approaches in this set. The Nyström method approximates the kernel matrix by sampling a subset of its columns. It was adopted recently by Yu et al. [277] for kernel CCA. Meanwhile, random Fourier features was adopted by Wu et al. [279] for kernel PCA. This scheme exploits Bochner's theorem [59,279], in which the kernel mapping is approximated by passing the data through a randomized projection and cosine functions. This results to a map of lower dimensions which saves computational cost. For more information, see the theoretical and empirical comparison of the Nyström method and random Fourier features by Yang et al. [318]. Other related low-rank approximation schemes were proposed by Peng et al. [283] which applies to kernel ICA, and that of Zhou et al. [286] called randomized kernel PCA. Lastly, a different approximation using the Taylor expansion of the RBF kernel was also derived by Wang et al. [288,289], and was called kernel sample equivalent replacement.

#### 4.9. Manifold Learning and Local Structure Analysis

The kernel MSPM methods described thus far are limited in their ability to learn local structure. A famous example that exhibits local structure would be the S-curve data set, described in [319], which is a sheet of points forming an "S" in 3-D space (see Figure 9a). In this case, manifold learning methods are more appropriate for dimensionality reduction. While kernel PCA aims to preserve nonlinear global directions with the maximum variance, manifold learning methods are constrained to preserve the distances between data points in their local neighborhoods [320]. For the S-curve data, this means that manifold learning methods will be able to "unfold" the curve in a 2-D mapping so that the points from either end of the curve become farthest apart, whereas kernel PCA would undesirably map

them close together. In Figure 9c, local linear embedding (LLE) was used as the manifold learner. The concept of manifold learning, sometimes called *local structure analysis*, was already adopted by many process monitoring researchers, which we review as follows.



**Figure 9.** Illustration of manifold learning: (a) S-curve data set; (b) 2-D Kernel principal components analysis (PCA) projection using radial basis function (RBF) kernel,  $c = 10$ ; (c) 2-D Local linear embedding (LLE) using kNN,  $k = 15$ . See [319] for more details.

The first few efforts to learn nonlinear manifolds via kernels for process monitoring were done by Shao et al. [108,109] in 2009. The techniques in [108,109] are related to maximum variance unfolding (MVU), which is a variant of kernel PCA that does not require selecting a kernel function *a priori*. Instead, MVU automatically *learns* the kernel matrix from the training data [109,320]. However, a parameter for defining the neighborhood must still be adjusted, for instance, the number of nearest neighbors,  $k$ . The strategy in [109] is to set  $k$  as the smallest integer that makes the entire neighborhood graph fully connected. Shao and Rong [109] have shown that the spectrum of the kernel matrix from MVU reveals a sharper contrast between the dominant and non-dominant eigenvalues than that from kernel PCA for the TEP case study. This result is important as it indicates that the salient features were separated from the noise more effectively. Other than MVU, a more popular technique is locality preserving projections (LPP), originally proposed by He and Niyogi [321] and then adopted by Hu and Yuan [322] for batch process monitoring. MVU only computes an embedding for the training data, hence, it requires a regression step to find the explicit mapping function for any test data. In contrast, the explicit mapping is readily available for LPP. The kernel version of LPP was adopted by Deng et al. [149,150] for process monitoring. Meanwhile, generalized LPP and discriminative LPP (and its kernel version) were proposed by Shao et al. [110] and Rong et al. [151], respectively. Other works that adopted variants of LPP can be found in [218,234,252,258,266,273,290]. The heat kernel (HK) is commonly used as a weighting function in LPP.

More recently, researchers have recognized that both global and local structure must be learned rather than focusing on one or the other. Hence, Luo et al. [182,187] proposed the kernel global-local preserving projections (GLPP). The projections from GLPP are in the middle of those from LPP and PCA because the local (LPP) and global (PCA) structures are simultaneously preserved. Other works in this regard can be found in [204,215,222,279,282]. To learn more about manifold learning, we refer the reader to a comparative review of dimensionality reduction methods by Van der Maaten et al. [320]. The connection between manifold learning and kernel PCA is also discussed by Ham et al. [323].

#### 4.10. Time-Varying Behavior and Adaptive Kernel Computation

When an MSPM method is successfully trained and deployed for process monitoring, it is usually assumed that the normal process behavior represented in the training data is the same behavior to be monitored during the testing phase. This means that the computed projection matrices and upper control limits (UCLs) are fixed or *time-invariant*. However, in practice, the process behavior continuously changes. Even if sophisticated detection models were used, a changing process behavior would require the model to be *adaptive*. That is, the model must adapt to changes in the normal

behavior without accommodating any fault behavior. However, it would be time-consuming for the model to be re-trained from scratch every time a new sample arrives. Hence, a recurrence relation or a recursive scheme must be formulated to make the model adaptive. For kernel methods, the actual issue is that kernel matrix adaptation is not straightforward. As noted by Hoegaerts et al. [324], adapting a linear PCA covariance matrix to a new data point will not change its size, whereas doing so for a kernel matrix would expand both its row and column dimensions. Hence, to keep its size, the kernel matrix must be *updated* and *downdated* at the same time. In addition, the eigendecomposition of the kernel matrix must also be adapted, wherein the number of retained principal components may change. These notions are important for addressing the time-varying process behavior.

In 2009, Liu et al. [111] proposed a moving window kernel PCA by implementing the adaptive schemes from Hoegaerts et al. [324] and Hall et al. [325]. It was applied to a butane distillation process where the fresh feed flow and the fresh feed temperature are time-varying. During implementation, adaptive control charts were produced, where the UCLs vary with time and the number of retained principal components varied between 8 and 13 as well. Khediri et al. [126] then proposed a variable moving window scheme where the model can be updated with a block of new data instead of a single data point. Meanwhile, Jaffel et al. [191] proposed a moving window reduced kernel PCA, where “reduced” pertains to an approach for easing the computational burden as discussed in Section 4.8. Other related works that utilize the moving window concept can be found in [190,207–209,238,293]. A different adaptive approach is to use multivariate EWMA to update any part of the model, such as the kernel matrix, its eigen-decomposition, or the statistical indices [116,132,179,224,253,281,283,292]. Finally, for the dictionary learning approach by Fezai et al. [246,247] (see Section 4.8), the Woodbury matrix identity is required to update the inverse of the kernel matrix, thereby updating the dictionary of kernel features as well. This scheme was adopted later in [250,270,271].

#### 4.11. Multi-Block and Distributed Monitoring

Due to the enormous scale of industrial plants nowadays, having a centralized process monitoring system for the entire plant has its limitations. According to Jiang and Huang [326], a centralized system may be limited in terms of: (1) fault-tolerance—it may fail to recognize faults if many of them occur simultaneously at different locations; (2) reliability—because it handles all data channels, it is more likely to fail if ever one of the channels become unavailable; (3) economic efficiency—it does not account for geographically distant process units that should naturally be monitored separately; and (4) performance—its monitoring performance can still be improved by decomposing the plant into blocks. These reasons have led to the rise of multi-block, distributed, or decentralized process monitoring methods, of which the kernel-based ones are reviewed as follows.

Kernel PLS is widely applied to decentralized process monitoring, as found in [101,119,129,206,284]. Lu and Wang [101] utilized a signed digraph, which was mentioned in Section 4.3.3 to have achieved fault diagnosis by incorporating causality. Zhang et al. [119] proposed the multi-block kernel PLS to monitor the continuous annealing process (CAP) case study, and utilized the fact that each of the 18 rolls in the process constitute a block of variables. By monitoring each of the 18 blocks rather than the entire process as one, it becomes easier to diagnose the fault location. An equivalent multi-block multi-scale kernel PLS was used by Zhang and Hu [129] in the PenSim and the electro-fused magnesia furnace (EFMF) case studies. Multi-block kernel ICA was proposed by Zhang and Ma [133] to monitor the CAP case study as well. Enhanced results for the CAP was achieved by Liu et al. [241] by using dynamic concurrent kernel CCA with multi-block analysis for fault isolation. Peng et al. [283] also used a prior process knowledge of the TEP to partition the 33 process variables into 3 sub-blocks, each monitored by adaptive dynamic kernel ICA.

In order to perform block division when process knowledge is not available, Jiang and Yan [327] proposed to use mutual information (MI) based clustering. This idea was fused with kernel PCA based process monitoring by Jiang and Yan [180], Huang and Yan [245], and Deng et al. [287]. All these works have used the TEP as a case study, and they have consistently elucidated 4 sub-blocks for the TEP.

For instance, in [245], their method initially produced 12 sub-blocks of variables, but 7 of these contain only one variable. Hence, some sub-blocks were fused into others, yielding only 4 sub-blocks in the end. Another approach is to divide the process according to blocks that give optimal fault detection performance, as proposed by Jiang et al. [198]. They used the genetic algorithm and kernel PCA for optimization and performance evaluation, respectively. Different from the above, Cai et al. [181] used kernel CCA to model the plant as a complex network and then used PCA for process monitoring. Li et al. [80,267] also proposed a hierarchical process modelling concept that separates the monitoring of linear from nonlinearly related variables. More recently, Yan et al. [284] used self-organizing maps (SOM) for block division, where the quality-related variables are monitored by kernel PLS and the quality-unrelated variables by kernel PCA.

For a systematic review of plant-wide monitoring methods, the reader can refer to Ge [33].

#### 4.12. Advanced Methods: Ensembles and Deep Learning

Ensemble learning and deep learning are two emerging concepts that have now become standard in the AI community [40]. The idea of ensemble learning is to build an enhanced model by combining the strengths of many simpler models [308]. The case for using ensembles is strong due to the many data science competitions that were won by exploiting the concept. For example, the winner of the Netflix Prize for a video recommender system was an ensemble of more than 100 learners [40], the winner of the Higgs Boson machine learning challenge was an ensemble of 70 deep neural networks that differ in initialization and training data sets [328], and it was reported that 17 out of the 29 challenges published in a machine learning competition site called Kaggle in 2015 alone were won by an ensemble learner called XGBoost [329]. Meanwhile, deep learning methods are general-purpose learning procedures for the automatic extraction of features using a multi-layer stack of input-output mappings [52]. Because features are learned automatically, it then avoids the task of designing feature extractors by hand, which would have required domain expertise. The case for using deep learners is strengthened by the fact that they have beaten many records in computer vision tasks, natural language processing tasks, video games, etc. [52,330]. In the process monitoring community, ensemble and deep architectures have also started appearing among kernel-based methods.

In 2015, Li and Yang [167] proposed an ensemble kernel PCA strategy wherein the base learners are kernel PCA models of various RBF kernel widths. For the TEP, 11 base models of kernel widths  $c = 2^{i-1}5m, i = 1, \dots, 11$  were used and gave better detection rates than using a single RBF kernel alone. Later on, Deng et al. [220] proposed Deep PCA by stacking together linear PCA and kernel PCA mappings. Bayesian inference was used to consolidate the monitoring statistics from each layer, so that a single final result is obtained. Using the TEP as case study, the detection rates of a 2-layer Deep PCA model were shown to have improved against linear PCA and kernel PCA alone. Further work in [256] used more layers in Deep PCA, as well as the FVS scheme (see Section 4.8) for reducing the computational cost. Deng et al. [257] also proposed serial PCA, where kernel PCA is performed on the residual space of an initial linear PCA transformation. In that work, the similarity factors method was used for fault classification as well (see Section 4.3.2). A different way to hybridize PCA and kernel PCA is by parallel instead of serial means, as proposed by Jiang and Yan [261]. Meanwhile, Li et al. [80,267] also used multi-level hierarchical models involving both linear PCA and kernel PCA. More recently, the ensemble kernel PCA was fused with local structure analysis by Cui et al. [273] for manifold learning (see Section 4.9).

We refer the reader to Lee et al. [9] for a more general outlook of the implications of advanced learning models to the process systems engineering field.

## 5. A Future Outlook on Kernel-Based Process Monitoring

Despite the many advances in kernel-based process monitoring research, more challenges are still emerging. It is likely that kernel methods, and other machine learning tools, as presented in Figure 4,

will have a role in addressing these challenges towards safer operations in the industry. A few of these challenges are discussed as follows.

### 5.1. Handling Heterogeneous and Multi-Rate Data

As introduced in Section 2, plant data sets are said to consist of  $N$  samples of  $M$  process variables. However, process measurements are not the only source of plant data. To perform process monitoring more effectively, it can also benefit from image data analytics, video data analytics, and alarm analytics. One notable work by Feng et al. [262] used kernel ICA to analyze video information for process monitoring. A more recent integration of alarm analytics to fault detection and identification was also developed by Lucke et al. [331]. Aside from these, spectroscopic data could be another information source from the plant since it is used for elucidating chemical structure. In addition, process monitoring can also be improved by combining information from both low- and high-frequency process measurements. Most of the case studies in the papers reviewed here generate only low-frequency data, e.g., 3-s sampling interval for the TEP. But there also exist data from pressure transducers (5 kHz), vibration measurements (0.5 Hz–10 kHz), and so on. Ruiz-Carcel et al. [332], for instance, have combined these multi-rate data to perform fault detection and diagnosis using CVA. It is projected that more efforts to handle heterogeneous and multi-rate data will appear in the future.

Although the above issues are recognized, the way to move forward is to first establish benchmark case studies that exhibit heterogeneous and multi-rate data. This will help ensure that new methods for handling these issues can be fairly compared. One such data set has been generated and made publicly available by Stief et al. [333], namely, from a real-world multiphase flow facility. For more details about the data set and how to acquire it, see the above reference.

### 5.2. Performing Fault Prognosis

Fault detection and diagnosis are the main objectives of the papers found in this review. As noted in Section 1, the third component of process monitoring is fault prognosis. After detecting and localizing the fault, prognosis methods aim to predict the future behavior of the process under faulty conditions. If the fault would lead to process failure, it is important to know in advance when it would happen, along with a measure of its uncertainty. This quantity is known as the *remaining useful life* or *time-to-failure* of the process [334]. Once these quantities are computed, the appropriate maintenance or repair actions can be performed, and hence, failure or emergency situations can be prevented.

To perform prognosis, the first step is to extract an incipient fault signal from the measured variables that is separated from noise and other disturbances as clearly as possible. This means that the method used for feature extraction should handle the incipient fault detection issue very well (see Section 4.5). Secondly, the drifting behavior of the incipient fault must be extrapolated into the future using a predictive model. This predictive element is key to the prognosis performance. The model must have a satisfactory extrapolation ability, that is, the ability to make reliable predictions beyond the data space where it was initially trained [20]. For instance, a detection model based on the widely used RBF kernel would have poor extrapolation abilities, as noted in Pilario et al. [67]. To solve this, a mixture of the RBF and the POLY kernels was used to improve both interpolation and extrapolation abilities. These kernels were adopted into kernel CVA for incipient fault monitoring. Another kernel method for prediction is Gaussian Processes (GP), which was used by Ge [335] under the PCA framework. Also, Ma et al. [265] used the fault reconstruction approach in kernel ICA to generate fault signals for prediction. Meanwhile, Xu et al. [186] used a neural network for prediction, together with local kernel PCA based monitoring.

Despite these efforts, predictive tasks are generally considered difficult, especially in nonlinear dynamic processes. For *nonlinear* processes, predictions will be inaccurate if the hypothesis space of the assumed predictive model is not sufficient to capture the complex process behavior. And even if the hypothesis space is sufficient, enough training data must be acquired to search the correct model within the hypothesis space. However, training data is scarce during the initial stage of process degradation.

In other words, it is difficult to determine whether the future trend would be linear, exponential, or any other shape on the basis of only a few degradation samples. Furthermore, a process is *dynamic* if its behavior at one point in time depends on its behavior at a previous time. This means that if the current prediction is fed into a dynamic model to serve as input for the next prediction, then small errors will accumulate as predictions are made farther into the future. It is important to be aware of these issues when developing fault prognosis strategies for industrial processes.

### 5.3. Developing More Advanced Methods and Improving Kernel Designs

Due to the recent advances in AI research, more and more process monitoring methods that rely on ensembles and deep architectures are expected to appear in the future (see Section 4.12). As mentioned in Section 2.3, both kernel methods and deep ANNs can be exploited, possibly in combined form, in order to create more expressive models. In addition, more creative kernel designs can be used, especially via the *multiple kernel learning* approach as noted in [67,163,277]. Multiple kernels can be created by combining single kernels additively or multiplicatively while still satisfying Mercer's conditions [44,306]. The combination of kernels can be done in series, in parallel, or both. For instance, the proposed serial PCA [257] and deep PCA [220] architectures can pave the way for deep kernel learning for process monitoring. Also, the concept of automatic relevance determination [314] can be considered in future works, wherein the Gaussian kernel width is allowed to have different values in each dimension of the data space. New kernel designs can also be inspired by the challenge of handling heterogeneous data, as mentioned in Section 5.1. Many examples of kernel designs for other types of data have already been used [44], such as for strings of text, images, gene expressions (bioinformatics), and categorical data. Hence, new kernel designs for heterogeneous process data may be inspired by these examples.

In parallel with these developments, a more careful approach to kernel parameter selection must be carried out, such as cross-validation and optimization techniques. To ensure that new results can be replicated and verified, we encourage researchers to always state the kernel functions chosen, the kernel parameter selection route, and how all other settings were obtained in their methods. The repeatability of results strengthens the understanding of new concepts, which will further lead to newer concepts more quickly. Hence, these efforts are necessary to further the development of the next generation of methods for fault detection, fault diagnosis, and fault prognosis in industrial plants.

It is important to note, however, that the development of new methods must be driven by the needs of the industry rather than for the sake of simply implementing new techniques. This means that, although it is tempting to develop a sophisticated method that can handle all the issues discussed in this article, it is more beneficial to understand the case study and the characteristics of the plant data at hand so that the right solutions are delivered to the end users.

## 6. Conclusions

In this paper, we reviewed the applications of kernel methods to perform nonlinear process monitoring. This paper firstly discussed the relationship between kernel methods and other techniques from machine learning, more importantly neural networks. Within this context, we gave motivations on why kernel methods are worthwhile to consider to perform nonlinear feature extraction from industrial plant data.

Based on 230 collected papers from 2004 to 2019, this article then identified 12 major issues that researchers aim to address regarding the use of kernel methods as feature extractors. We discussed issues such as how to choose the kernel function, how to decide kernel parameters, how to perform fault diagnosis in kernel feature space, how to compute kernel mappings faster, how to make the kernel computation adaptive, how to learn manifolds or local structures, and how to benefit from ensembles and deep architectures. The rest of the topics include how to handle batch process data, how to account for process dynamics, how to monitor quality variables, how to improve detection, and how to distribute the monitoring task across the whole plant. By addressing these issues, we have seen

how nonlinear process monitoring research has progressed extensively in the last 15 years, through the impact of kernel methods.

Finally, potential future directions on kernel-based process monitoring research were presented. Emerging topics on new kernel designs, handling heterogeneous data, and performing fault prognosis were deemed worthwhile to investigate. In order to move forward, we encourage more researchers to venture in this area of process monitoring. For interested readers, this article is also supplemented by MATLAB codes for SVM and kernel PCA (see Figure 3 and Ref. [315]), which were made available to the public. We hope that this article can contribute to the further understanding of the role of kernel methods in process monitoring, and provide new insights for researchers in the field.

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

**Funding:** This research is supported by the Faculty Development Fund of the Engineering Research and Development for Technology (ERDT) program of the Department of Science and Technology (DOST), Philippines. Support from the National Key Research and Development Plan (2018YFC0214102) of P. R. China is also acknowledged.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## Abbreviations

The following abbreviations are used in the manuscript text:

AI	Artificial Intelligence	MI	Mutual Information
ANN	Artificial Neural Network	MSPM	Multivariate Statistical Process Monitoring
CNKI	China National Knowledge Infrastructure	PSE	Process Systems Engineering
DTW	Dynamic Time Warping	RKHS	Reproducing Kernel Hilbert Space
EWMA	Exponentially Weighted Moving Average	SCADA	Supervisory Control and Data Acquisition
FVS	Feature Vector Selection	SDG	Signed Digraph
GA	Genetic Algorithm	SFM	Similarity Factor Method
GLRT	Generalized Likelihood Ratio Test	SOM	Self-organizing Maps
GP	Gaussian Processes	SPA	Statistical Pattern Analysis
KDE	Kernel Density Estimation	SVDD	Support Vector Data Description
kNN	<i>k</i> -Nearest Neighbors	SVM	Support Vector Machine
KPCA	Kernel Principal Components Analysis	UCL	Upper Control Limit

Abbreviations of the kernelized methods in Table 3 are as follows:

AMD	Augmented Mahalanobis distance	ICA	Independent components analysis
C-PLS	Concurrent partial least squares	K-means	K-means clustering
CCA	Canonical correlation analysis	LLE	Local linear embedding
CVA	Canonical variate analysis	LPP	Locality preserving projections
DD	Direct decomposition	LS	Least squares
DISSIM	Dissimilarity analysis	MVU	Maximum variance unfolding
DL	Dictionary learning	NNMF	Non-negative matrix factorization
DLV	Dynamic latent variable model	PCA	Principal components analysis
ECA	Entropy components analysis	PCR	Principal component regression
EDA	Exponential discriminant analysis	PLS	Partial least squares
ELM	Extreme learning machine	RPLVR	Robust probability latent variable regression
FDA	Fisher discriminant analysis	SDA	Scatter-difference-based discriminant analysis
DFDA	Fault-degradation-oriented FDA	SFA	Slow feature analysis
GLPP	Global-local preserving projections	T-PLS	Total partial least squares
GMM	Gaussian mixture model	VCA	Variable correlations analysis



Abbreviations of the case studies in Table 3 are as follows:

AEP	Aluminum electrolysis process	HGPWLTP	Hot galvanizing pickling waste liquor treatment process
AIRLOR	Air quality monitoring network		
BAFP	Biological anaerobic filter process	HSMP	Hot strip mill process
BDP	Butane distillation process	IGT	Industrial gas turbine
CAP	Continuous annealing process	IMP	Injection moulding process
CFPP	Coal-fired power plant	IPOP	Industrial p-xylene oxidation process
CLG	Cyanide leaching of gold	MFF	Multiphase flow facility
CPP	Cigarette production process	NE	Numerical example
CSEC	Cad System in E. coli	NPP	Nosiheptide production process
CSTH	Continuous stirred-tank heater	PCBP	Polyvinyl chloride batch process
CSTR	Continuous stirred-tank reactor	PenSim	Penicillin fermentation process
DMCP	Dense medium coal preparation	PP	Polymerization process
DP	Drying process	PV	Photovoltaic systems
DTS	Dissolution tank system	RCP	Real chemical process
EFMF	Electro-fused magnesia furnace	SEP	Semiconductor etch process
FCCU	Fluid catalytic cracking unit	TEP	Tennessee Eastman plant
GCND	Genomic copy number data	TPP	Thermal power plant
GHP	Gold hydrometallurgy process	TTP	Three-tank process
GMP	Glass melter process	WWTP	Wastewater treatment plant

Abbreviations of kernel functions in Table 3 are as follows:

RBF	Gaussian radial basis function kernel	HK	Heat kernel
POLY	Polynomial kernel	SIG	Sigmoid kernel
COS	Cosine kernel	NSDC	Non-stationary discrete convolution kernel
WAV	Wavelet kernel		

## References

- Chiang, L.H.; Russell, E.L.; Braatz, R.D. *Fault Detection and Diagnosis in Industrial Systems*; Springer: London, UK, 2005; pp. 1–280.
- Reis, M.; Gins, G. Industrial Process Monitoring in the Big Data/Industry 4.0 Era: From Detection, to Diagnosis, to Prognosis. *Processes* **2017**, *5*, 35. [\[CrossRef\]](#)
- Isermann, R. Model-based fault-detection and diagnosis—Status and applications. *Annu. Rev. Control* **2005**, *29*, 71–85. [\[CrossRef\]](#)
- Pilario, K.E.S.; Cao, Y.; Shafiee, M. Incipient Fault Detection, Diagnosis, and Prognosis using Canonical Variate Dissimilarity Analysis. *Comput. Aided Chem. Eng.* **2019**, *46*, 1195–1200. [\[CrossRef\]](#)
- Pilario, K.E.S.; Cao, Y. Canonical Variate Dissimilarity Analysis for Process Incipient Fault Detection. *IEEE Trans. Ind. Inform.* **2018**, *14*, 5308–5315. [\[CrossRef\]](#)
- Ge, Z.; Song, Z.; Gao, F. Review of recent research on data-based process monitoring. *Ind. Eng. Chem. Res.* **2013**, *52*, 3543–3562. [\[CrossRef\]](#)
- Venkatasubramanian, V. DROWNING IN DATA: Informatics and modeling challenges in a data-rich networked world. *AIChE J.* **2009**, *55*, 2–8. [\[CrossRef\]](#)
- Kourti, T. Process analysis and abnormal situation detection: From theory to practice. *IEEE Control Syst. Mag.* **2002**, *22*, 10–25. [\[CrossRef\]](#)
- Lee, J.H.; Shin, J.; Realff, M.J. Machine learning: Overview of the recent progresses and implications for the process systems engineering field. *Comput. Chem. Eng.* **2018**, *114*, 111–121. [\[CrossRef\]](#)
- Ning, C.; You, F. Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming. *Comput. Chem. Eng.* **2019**, *125*, 434–448. [\[CrossRef\]](#)
- Ge, Z.; Song, Z.; Ding, S.X.; Huang, B. Data Mining and Analytics in the Process Industry: The Role of Machine Learning. *IEEE Access* **2017**, *5*, 20590–20616. [\[CrossRef\]](#)
- Shu, Y.; Ming, L.; Cheng, F.; Zhang, Z.; Zhao, J. Abnormal situation management: Challenges and opportunities in the big data era. *Comput. Chem. Eng.* **2016**, *91*, 104–113. [\[CrossRef\]](#)

13. Chiang, L.; Lu, B.; Castillo, I. Big Data Analytics in Chemical Engineering. *Annu. Rev. Chem. Biomol. Eng.* **2017**, *8*, 63–85. [[CrossRef](#)] [[PubMed](#)]
14. Venkatasubramanian, V. The promise of artificial intelligence in chemical engineering: Is it here, finally? *AIChE J.* **2019**, *65*, 466–478. [[CrossRef](#)]
15. Qin, S.J. Process Data Analytics in the Era of Big Data. *AIChE J.* **2014**, *60*, 3092–3100. [[CrossRef](#)]
16. Qin, S.J.; Chiang, L.H. Advances and opportunities in machine learning for process data analytics. *Comput. Chem. Eng.* **2019**, *126*, 465–473. [[CrossRef](#)]
17. Patwardhan, R.S.; Hamadah, H.A.; Patel, K.M.; Hafiz, R.H.; Al-Gwaiz, M.M. Applications of Advanced Analytics at Saudi Aramco: A Practitioners' Perspective. *Ind. Eng. Chem. Res.* **2019**, *58*, 11338–11351. [[CrossRef](#)]
18. Ruiz-Cárcel, C.; Cao, Y.; Mba, D.; Lao, L.; Samuel, R.T. Statistical process monitoring of a multiphase flow facility. *Control Eng. Pract.* **2015**, *42*, 74–88. [[CrossRef](#)]
19. Ruiz-Cárcel, C.; Lao, L.; Cao, Y.; Mba, D. Canonical variate analysis for performance degradation under faulty conditions. *Control Eng. Pract.* **2016**, *54*, 70–80. [[CrossRef](#)]
20. Nelles, O. *Nonlinear System Identification*; Springer: London, UK, 2001; p. 785. [[CrossRef](#)]
21. Cover, T.M. Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition. *IEEE Trans. Electron. Comput.* **1965**, *EC-14*, 326–334. [[CrossRef](#)]
22. Schölkopf, B.; Smola, A.; Müller, K.R. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput.* **1998**, *10*, 1299–1319. [[CrossRef](#)]
23. Max Planck Society. 2019. Available online: <https://www.mpg.de/13645470/schoelkopf-koerber-prize> (accessed on 30 October 2019).
24. Lee, J.M.; Yoo, C.; Choi, S.W.; Vanrolleghem, P.A.; Lee, I.B. Nonlinear process monitoring using kernel principal component analysis. *Chem. Eng. Sci.* **2004**, *59*, 223–234. [[CrossRef](#)]
25. Qin, S.J. Survey on data-driven industrial process monitoring and diagnosis. *Annu. Rev. Control* **2012**, *36*, 220–234. [[CrossRef](#)]
26. MacGregor, J.; Cinar, A. Monitoring, fault diagnosis, fault-tolerant control and optimization: Data driven methods. *Comput. Chem. Eng.* **2012**, *47*, 111–120. [[CrossRef](#)]
27. Yin, S.; Ding, S.X.; Xie, X.; Luo, H. A review on basic data-driven approaches for industrial process monitoring. *IEEE Trans. Ind. Electron.* **2014**, *61*, 6414–6428. [[CrossRef](#)]
28. Ding, S.X. Data-driven design of monitoring and diagnosis systems for dynamic processes: A review of subspace technique based schemes and some recent results. *J. Process Control* **2014**, *24*, 431–449. [[CrossRef](#)]
29. Yin, S.; Li, X.; Gao, H.; Kaynak, O. Data-based techniques focused on modern industry: An overview. *IEEE Trans. Ind. Electron.* **2015**, *62*, 657–667. [[CrossRef](#)]
30. Severson, K.; Chaiwatanodom, P.; Braatz, R.D. Perspectives on process monitoring of industrial systems. *Annu. Rev. Control* **2016**, *42*, 190–200. [[CrossRef](#)]
31. Tidriri, K.; Chatti, N.; Verron, S.; Tiplica, T. Bridging data-driven and model-based approaches for process fault diagnosis and health monitoring: A review of researches and future challenges. *Annu. Rev. Control* **2016**, *42*, 63–81. [[CrossRef](#)]
32. Yin, Z.; Hou, J. Recent advances on SVM based fault diagnosis and process monitoring in complicated industrial processes. *Neurocomputing* **2016**, *174*, 643–650. [[CrossRef](#)]
33. Ge, Z. Review on data-driven modeling and monitoring for plant-wide industrial processes. *Chemom. Intell. Lab. Syst.* **2017**, *171*, 16–25. [[CrossRef](#)]
34. Wang, Y.; Si, Y.; Huang, B.; Lou, Z. Survey on the theoretical research and engineering applications of multivariate statistics process monitoring algorithms: 2008–2017. *Can. J. Chem. Eng.* **2018**, *96*, 2073–2085. [[CrossRef](#)]
35. Md Nor, N.; Che Hassan, C.R.; Hussain, M.A. A review of data-driven fault detection and diagnosis methods: Applications in chemical process systems. *Rev. Chem. Eng.* **2018**. [[CrossRef](#)]
36. Alauddin, M.; Khan, F.; Imtiaz, S.; Ahmed, S. A Bibliometric Review and Analysis of Data-Driven Fault Detection and Diagnosis Methods for Process Systems. *Ind. Eng. Chem. Res.* **2018**, *57*, 10719–10735. [[CrossRef](#)]
37. Jiang, Q.; Yan, X.; Huang, B. Review and Perspectives of Data-Driven Distributed Monitoring for Industrial Plant-Wide Processes. *Ind. Eng. Chem. Res.* **2019**, *58*, 12899–12912. [[CrossRef](#)]

38. Quiñones-Grueiro, M.; Prieto-Moreno, A.; Verde, C.; Llanes-Santiago, O. Data-driven monitoring of multimode continuous processes: A review. *Chemom. Intell. Lab. Syst.* **2019**, *189*, 56–71. [[CrossRef](#)]
39. Wang, S.; Aggarwal, C.; Liu, H. Randomized Feature Engineering as a Fast and Accurate Alternative to Kernel Methods. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining—KDD'17, Halifax, NS, Canada, 13–17 August 2017; pp. 485–494. [[CrossRef](#)]
40. Domingos, P. A few useful things to know about machine learning. *Commun. ACM* **2012**, *55*, 78. [[CrossRef](#)]
41. Vert, J.; Tsuda, K.; Schölkopf, B. A primer on kernel methods. *Kernel Methods Comput. Biol.* **2004**, 35–70. [[CrossRef](#)]
42. Cao, D.S.; Liang, Y.Z.; Xu, Q.S.; Hu, Q.N.; Zhang, L.X.; Fu, G.H. Exploring nonlinear relationships in chemical data using kernel-based methods. *Chemom. Intell. Lab. Syst.* **2011**, *107*, 106–115. [[CrossRef](#)]
43. Shawe-Taylor, J.; Cristianini, N. *Kernel Methods for Pattern Analysis*; Cambridge University Press: New York, NY, USA, 2004.
44. Cristianini, N.; Shawe-Taylor, J. *Support Vector Machines and Other Kernel-based Learning Methods*; Cambridge University Press: Cambridge, UK, 2014.
45. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2006.
46. Kolesnikov, A.; Zhai, X.; Beyer, L. Revisiting Self-Supervised Visual Representation Learning. *arXiv* **2019**, arXiv:1901.09005.
47. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; MIT Press: Cambridge, MA, USA, 2012.
48. Dong, D.; McAvoy, T. Nonlinear principal component analysis based on principal curves and neural networks. *Comput. Chem. Eng.* **1996**, *20*, 65–78. [[CrossRef](#)]
49. Hornik, K.; Stinchcombe, M.; White, H. Multilayer Feedforward Networks are Universal Approximators. *Neural Netw.* **1989**, *2*, 359–366. [[CrossRef](#)]
50. Domingos, P. *The Master Algorithm*; Basic Books: New York City, NY, USA, 2015.
51. Belkin, M.; Ma, S.; Mandal, S. To understand deep learning we need to understand kernel learning. In Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; Volume 2, pp. 874–882.
52. Lecun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)] [[PubMed](#)]
53. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *Proc. IEEE Int. Conf. Comput. Vis.* **2015**, *2015*, 1026–1034. [[CrossRef](#)]
54. Huang, P.S.; Avron, H.; Sainath, T.N.; Sindhvani, V.; Ramabhadran, B. Kernel methods match Deep Neural Networks on TIMIT. In Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Florence, Italy, 4–9 May 2014; pp. 205–209. [[CrossRef](#)]
55. Skrimpas, G.A.; Sweeney, C.W.; Marhadi, K.S.; Jensen, B.B.; Mijatovic, N.; Holbøll, J. Employment of kernel methods on wind turbine power performance assessment. *IEEE Trans. Sustain. Energy* **2015**, *6*, 698–706. [[CrossRef](#)]
56. Song, C.; Liu, K.; Zhang, X. Integration of Data-Level Fusion Model and Kernel Methods for Degradation Modeling and Prognostic Analysis. *IEEE Trans. Reliab.* **2018**, *67*, 640–650. [[CrossRef](#)]
57. Eyo, E.N.; Pilario, K.E.S.; Lao, L.; Falcone, G. Development of a Real-Time Objective Gas—Liquid Flow Regime Identifier Using Kernel Methods. *IEEE Trans. Cybern.* **2019**, 1–11. [[CrossRef](#)]
58. Sun, S.; Zhang, G.; Wang, C.; Zeng, W.; Li, J.; Grosse, R. Differentiable compositional kernel learning for Gaussian processes. In Proceedings of the 35th International Conference on Machine Learning (ICML 2018), Stockholm, Sweden, 10–15 July 2018; Volume 11, pp. 7676–7696.
59. Mehrkanoon, S.; Suykens, J.A. Deep hybrid neural-kernel networks using random Fourier features. *Neurocomputing* **2018**, *298*, 46–54. [[CrossRef](#)]
60. Mehrkanoon, S. Deep neural-kernel blocks. *Neural Netw.* **2019**, *116*, 46–55. [[CrossRef](#)]
61. Wilson, A.G.; Hu, Z.; Salakhutdinov, R.; Xing, E.P. Deep Kernel Learning. *Mach. Learn.* **2015**, *72*, 1508–1524.
62. Wilson, A.G.; Hu, Z.; Salakhutdinov, R.; Xing, E.P. Stochastic variational deep kernel learning. In Proceedings of the Advances in Neural Information Processing Systems 29 (NIPS 2016), Barcelona, Spain, 5–10 December 2016; pp. 2594–2602.
63. Liu, Y.; Yang, C.; Gao, Z.; Yao, Y. Ensemble deep kernel learning with application to quality prediction in industrial polymerization processes. *Chemom. Intell. Lab. Syst.* **2018**, *174*, 15–21. [[CrossRef](#)]
64. Heng, A.; Zhang, S.; Tan, A.C.C.; Mathew, J. Rotating machinery prognostics: State of the art, challenges and opportunities. *Mech. Syst. Signal Process.* **2009**, *23*, 724–739. [[CrossRef](#)]

65. Kan, M.S.; Tan, A.C.C.; Mathew, J. A review on prognostic techniques for non-stationary and non-linear rotating systems. *Mech. Syst. Signal Process.* **2015**, *62*, 1–20. [[CrossRef](#)]
66. Fan, J.; Wang, Y. Fault detection and diagnosis of non-linear non-Gaussian dynamic processes using kernel dynamic independent component analysis. *Inf. Sci.* **2014**, *259*, 369–379. [[CrossRef](#)]
67. Pilario, K.E.S.; Cao, Y.; Shafiee, M. Mixed kernel canonical variate dissimilarity analysis for incipient fault monitoring in nonlinear dynamic processes. *Comput. Chem. Eng.* **2019**, *123*, 143–154. [[CrossRef](#)]
68. Fu, Y.; Kruger, U.; Li, Z.; Xie, L.; Thompson, J.; Rooney, D.; Hahn, J.; Yang, H. Cross-validatory framework for optimal parameter estimation of KPCA and KPLS models. *Chemom. Intell. Lab. Syst.* **2017**, *167*, 196–207. [[CrossRef](#)]
69. Vitale, R.; de Noord, O.E.; Ferrer, A. A kernel-based approach for fault diagnosis in batch processes. *J. Chemom.* **2014**, *28*, 697–707. [[CrossRef](#)]
70. Chiang, L.H.; Leardi, R.; Pell, R.J.; Seasholtz, M.B. Industrial experiences with multivariate statistical analysis of batch process data. *Chemom. Intell. Lab. Syst.* **2006**, *81*, 109–119. [[CrossRef](#)]
71. Lee, J.M.; Yoo, C.K.; Lee, I.B. Fault detection of batch processes using multiway kernel principal component analysis. *Comput. Chem. Eng.* **2004**, *28*, 1837–1847. [[CrossRef](#)]
72. Zhang, Y.; Qin, S.J. Fault detection of nonlinear processes using multiway kernel independent component analysis. *Ind. Eng. Chem. Res.* **2007**, *46*, 7780–7787. [[CrossRef](#)]
73. Tian, X.; Zhang, X.; Deng, X.; Chen, S. Multiway kernel independent component analysis based on feature samples for batch process monitoring. *Neurocomputing* **2009**, *72*, 1584–1596. [[CrossRef](#)]
74. Cho, H.W. Nonlinear feature extraction and classification of multivariate data in kernel feature space. *Expert Syst. Appl.* **2007**, *32*, 534–542. [[CrossRef](#)]
75. Yu, J. Nonlinear bioprocess monitoring using multiway kernel localized fisher discriminant analysis. *Ind. Eng. Chem. Res.* **2011**, *50*, 3390–3402. [[CrossRef](#)]
76. Zhang, Y.; An, J.; Li, Z.; Wang, H. Modeling and monitoring for handling nonlinear dynamic processes. *Inf. Sci.* **2013**, *235*, 97–105. [[CrossRef](#)]
77. Tang, X.; Li, Y.; Xie, Z. Phase division and process monitoring for multiphase batch processes with transitions. *Chemom. Intell. Lab. Syst.* **2015**, *145*, 72–83. [[CrossRef](#)]
78. Birol, G.; Ündey, C.; Çinar, A. A modular simulation package for fed-batch fermentation: Penicillin production. *Comput. Chem. Eng.* **2002**, *26*, 1553–1565. [[CrossRef](#)]
79. Peng, K.; Zhang, K.; Li, G.; Zhou, D. Contribution rate plot for nonlinear quality-related fault diagnosis with application to the hot strip mill process. *Control Eng. Pract.* **2013**, *21*, 360–369. [[CrossRef](#)]
80. Li, W.; Zhao, C. Hybrid fault characteristics decomposition based probabilistic distributed fault diagnosis for large-scale industrial processes. *Control Eng. Pract.* **2019**, *84*, 377–388. [[CrossRef](#)]
81. Vitale, R.; de Noord, O.E.; Ferrer, A. Pseudo-sample based contribution plots: Innovative tools for fault diagnosis in kernel-based batch process monitoring. *Chemom. Intell. Lab. Syst.* **2015**, *149*, 40–52. [[CrossRef](#)]
82. Rashid, M.M.; Yu, J. Nonlinear and non-Gaussian dynamic batch process monitoring using a new multiway kernel independent component analysis and multidimensional mutual information based dissimilarity approach. *Ind. Eng. Chem. Res.* **2012**, *51*, 10910–10920. [[CrossRef](#)]
83. Peng, C.; Qiao, J.; Zhang, X.; Lu, R. Phase Partition and Fault Diagnosis of Batch Process Based on KECA Angular Similarity. *IEEE Access* **2019**, *7*, 125676–125687. [[CrossRef](#)]
84. Jia, M.; Xu, H.; Liu, X.; Wang, N. The optimization of the kind and parameters of kernel function in KPCA for process monitoring. *Comput. Chem. Eng.* **2012**, *46*, 94–104. [[CrossRef](#)]
85. Choi, S.W.; Lee, I.B. Nonlinear dynamic process monitoring based on dynamic kernel PCA. *Chem. Eng. Sci.* **2004**, *59*, 5897–5908. [[CrossRef](#)]
86. Choi, S.W.; Lee, C.; Lee, J.M.; Park, J.H.; Lee, I.B. Fault detection and identification of nonlinear processes based on kernel PCA. *Chemom. Intell. Lab. Syst.* **2005**, *75*, 55–67. [[CrossRef](#)]
87. Cho, J.H.; Lee, J.M.; Wook Choi, S.; Lee, D.; Lee, I.B. Fault identification for process monitoring using kernel principal component analysis. *Chem. Eng. Sci.* **2005**, *60*, 279–288. [[CrossRef](#)]
88. Yoo, C.K.; Lee, I.B. Nonlinear multivariate filtering and bioprocess monitoring for supervising nonlinear biological processes. *Process Biochem.* **2006**, *41*, 1854–1863. [[CrossRef](#)]
89. Lee, D.S.; Lee, M.W.; Woo, S.H.; Kim, Y.J.; Park, J.M. Multivariate online monitoring of a full-scale biological anaerobic filter process using kernel-based algorithms. *Ind. Eng. Chem. Res.* **2006**, *45*, 4335–4344. [[CrossRef](#)]

90. Zhang, X.; Yan, W.; Zhao, X.; Shao, H. Nonlinear On-line Process Monitoring and Fault Detection Based on Kernel ICA. In Proceedings of the 2006 International Conference on Information and Automation, Colombo, Sri Lanka, 15–17 December 2006; Volume 1, pp. 222–227. [[CrossRef](#)]
91. Deng, X.; Tian, X. Multivariate Statistical Process Monitoring Using Multi-Scale Kernel Principal Component Analysis. *IFAC-PapersOnLine* **2006**, *6*, 108–113. [[CrossRef](#)]
92. Cho, H.W. Identification of contributing variables using kernel-based discriminant modeling and reconstruction. *Expert Syst. Appl.* **2007**, *33*, 274–285. [[CrossRef](#)]
93. Sun, R.; Tsung, F.; Qu, L. Evolving kernel principal component analysis for fault diagnosis. *Comput. Ind. Eng.* **2007**, *53*, 361–371. [[CrossRef](#)]
94. Choi, S.W.; Morris, J.; Lee, I.B. Nonlinear multiscale modelling for fault detection and identification. *Chem. Eng. Sci.* **2008**, *63*, 2252–2266. [[CrossRef](#)]
95. Tian, X.; Deng, X. A fault detection method using multi-scale kernel principal component analysis. In Proceedings of the 27th Chinese Control Conference, Kunming, China, 16–18 July 2008; pp. 25–29. [[CrossRef](#)]
96. Wang, T.; Wang, X.; Zhang, Y.; Zhou, H. Fault detection of nonlinear dynamic processes using dynamic kernel principal component analysis. In Proceedings of the 2008 7th World Congress on Intelligent Control and Automation, Chongqing, China, 25–27 June 2008; pp. 3009–3014. [[CrossRef](#)]
97. Lee, J.M.; Qin, S.J.; Lee, I.B. Fault Detection of Non-Linear Processes Using Kernel Independent Component Analysis. *Can. J. Chem. Eng.* **2008**, *85*, 526–536. [[CrossRef](#)]
98. Cui, P.; Li, J.; Wang, G. Improved kernel principal component analysis for fault detection. *Expert Syst. Appl.* **2008**, *34*, 1210–1219. [[CrossRef](#)]
99. Cui, J.; Huang, W.; Miao, M.; Sun, B. Kernel scatter-difference-based discriminant analysis for fault diagnosis. In Proceedings of the 2008 IEEE International Conference on Mechatronics and Automation, Takamatsu, Japan, 5–8 August 2008; pp. 771–774. [[CrossRef](#)]
100. Zhang, Y.; Qin, S.J. Improved nonlinear fault detection technique and statistical analysis. *AIChE J.* **2008**, *54*, 3207–3220. [[CrossRef](#)]
101. Lü, N.; Wang, X. Fault diagnosis based on signed digraph combined with dynamic kernel PLS and SVR. *Ind. Eng. Chem. Res.* **2008**, *47*, 9447–9456. [[CrossRef](#)]
102. He, X.B.; Yang, Y.P.; Yang, Y.H. Fault diagnosis based on variable-weighted kernel Fisher discriminant analysis. *Chemom. Intell. Lab. Syst.* **2008**, *93*, 27–33. [[CrossRef](#)]
103. Cho, H.W. An orthogonally filtered tree classifier based on nonlinear kernel-based optimal representation of data. *Expert Syst. Appl.* **2008**, *34*, 1028–1037. [[CrossRef](#)]
104. Li, J.; Cui, P. Kernel scatter-difference-based discriminant analysis for nonlinear fault diagnosis. *Chemom. Intell. Lab. Syst.* **2008**, *94*, 80–86. [[CrossRef](#)]
105. Li, J.; Cui, P. Improved kernel fisher discriminant analysis for fault diagnosis. *Expert Syst. Appl.* **2009**, *36*, 1423–1432. [[CrossRef](#)]
106. Zhang, Y. Enhanced statistical analysis of nonlinear processes using KPCA, KICA and SVM. *Chem. Eng. Sci.* **2009**, *64*, 801–811. [[CrossRef](#)]
107. Zhang, Y.; Zhang, Y. Complex process monitoring using modified partial least squares method of independent component regression. *Chemom. Intell. Lab. Syst.* **2009**, *98*, 143–148. [[CrossRef](#)]
108. Shao, J.D.; Rong, G.; Lee, J.M. Learning a data-dependent kernel function for KPCA-based nonlinear process monitoring. *Chem. Eng. Res. Des.* **2009**, *87*, 1471–1480. [[CrossRef](#)]
109. Shao, J.D.; Rong, G. Nonlinear process monitoring based on maximum variance unfolding projections. *Expert Syst. Appl.* **2009**, *36*, 11332–11340. [[CrossRef](#)]
110. Shao, J.D.; Rong, G.; Lee, J.M. Generalized orthogonal locality preserving projections for nonlinear fault detection and diagnosis. *Chemom. Intell. Lab. Syst.* **2009**, *96*, 75–83. [[CrossRef](#)]
111. Liu, X.; Kruger, U.; Littler, T.; Xie, L.; Wang, S. Moving window kernel PCA for adaptive monitoring of nonlinear processes. *Chemom. Intell. Lab. Syst.* **2009**, *96*, 132–143. [[CrossRef](#)]
112. Ge, Z.; Yang, C.; Song, Z. Improved kernel PCA-based monitoring approach for nonlinear processes. *Chem. Eng. Sci.* **2009**, *64*, 2245–2255. [[CrossRef](#)]
113. Zhao, C.; Wang, F.; Zhang, Y. Nonlinear process monitoring based on kernel dissimilarity analysis. *Control Eng. Pract.* **2009**, *17*, 221–230. [[CrossRef](#)]

114. Zhao, C.; Gao, F.; Wang, F. Nonlinear batch process monitoring using phase-based kernel-independent component analysis-principal component analysis (KICA-PCA). *Ind. Eng. Chem. Res.* **2009**, *48*, 9163–9174. [[CrossRef](#)]
115. Jia, M.; Chu, F.; Wang, F.; Wang, W. On-line batch process monitoring using batch dynamic kernel principal component analysis. *Chemom. Intell. Lab. Syst.* **2010**, *101*, 110–122. [[CrossRef](#)]
116. Cheng, C.Y.; Hsu, C.C.; Chen, M.C. Adaptive kernel principal component analysis (KPCA) for monitoring small disturbances of nonlinear processes. *Ind. Eng. Chem. Res.* **2010**, *49*, 2254–2262. [[CrossRef](#)]
117. Alcalá, C.F.; Qin, S.J. Reconstruction-Based Contribution for Process Monitoring with Kernel Principal Component Analysis. *Ind. Eng. Chem. Res.* **2010**, *49*, 7849–7857. [[CrossRef](#)]
118. Zhu, Z.B.; Song, Z.H. Fault diagnosis based on imbalance modified kernel Fisher discriminant analysis. *Chem. Eng. Res. Des.* **2010**, *88*, 936–951. [[CrossRef](#)]
119. Zhang, Y.; Zhou, H.; Qin, S.J.; Chai, T. Decentralized Fault Diagnosis of Large-Scale Processes Using Multiblock Kernel Partial Least Squares. *IEEE Trans. Ind. Inform.* **2010**, *6*, 3–10. [[CrossRef](#)]
120. Zhang, Y.; Li, Z.; Zhou, H. Statistical analysis and adaptive technique for dynamical process monitoring. *Chem. Eng. Res. Des.* **2010**, *88*, 1381–1392. [[CrossRef](#)]
121. Xu, J.; Hu, S. Nonlinear process monitoring and fault diagnosis based on KPCA and MKL-SVM. In Proceedings of the 2010 International Conference on Artificial Intelligence and Computational Intelligence, Sanya, China, 23–24 October 2010; Volume 1, pp. 233–237. [[CrossRef](#)]
122. Ge, Z.; Song, Z. Kernel generalization of PPCA for nonlinear probabilistic monitoring. *Ind. Eng. Chem. Res.* **2010**, *49*, 11832–11836. [[CrossRef](#)]
123. Wang, L.; Shi, H. Multivariate statistical process monitoring using an improved independent component analysis. *Chem. Eng. Res. Des.* **2010**, *88*, 403–414. [[CrossRef](#)]
124. Sumana, C.; Mani, B.; Venkateswarlu, C.; Gudi, R.D. Improved Fault Diagnosis Using Dynamic Kernel Scatter-Difference-Based Discriminant Analysis. *Ind. Eng. Chem. Res.* **2010**, *49*, 8575–8586. [[CrossRef](#)]
125. Sumana, C.; Bhushan, M.; Venkateswarlu, C.; Gudi, R.D. Improved nonlinear process monitoring using KPCA with sample vector selection and combined index. *Asia-Pac. J. Chem. Eng.* **2011**, *6*, 460–469. [[CrossRef](#)]
126. Khediri, I.B.; Limam, M.; Weihs, C. Variable window adaptive Kernel Principal Component Analysis for nonlinear nonstationary process monitoring. *Comput. Ind. Eng.* **2011**, *61*, 437–446. [[CrossRef](#)]
127. Zhang, Y.; Ma, C. Fault diagnosis of nonlinear processes using multiscale KPCA and multiscale KPLS. *Chem. Eng. Sci.* **2011**, *66*, 64–72. [[CrossRef](#)]
128. Zhang, Y.; Hu, Z. On-line batch process monitoring using hierarchical kernel partial least squares. *Chem. Eng. Res. Des.* **2011**, *89*, 2078–2084. [[CrossRef](#)]
129. Zhang, Y.; Hu, Z. Multivariate process monitoring and analysis based on multi-scale KPLS. *Chem. Eng. Res. Des.* **2011**, *89*, 2667–2678. [[CrossRef](#)]
130. Zhu, Z.B.; Song, Z.H. A novel fault diagnosis system using pattern classification on kernel FDA subspace. *Expert Syst. Appl.* **2011**, *38*, 6895–6905. [[CrossRef](#)]
131. Khediri, I.B.; Weihs, C.; Limam, M. Kernel k-means clustering based local support vector domain description fault detection of multimodal processes. *Expert Syst. Appl.* **2012**, *39*, 2166–2171. [[CrossRef](#)]
132. Zhang, Y.; Li, S.; Teng, Y. Dynamic processes monitoring using recursive kernel principal component analysis. *Chem. Eng. Sci.* **2012**, *72*, 78–86. [[CrossRef](#)]
133. Zhang, Y.; Ma, C. Decentralized fault diagnosis using multiblock kernel independent component analysis. *Chem. Eng. Res. Des.* **2012**, *90*, 667–676. [[CrossRef](#)]
134. Zhang, Y.; Li, S.; Hu, Z. Improved multi-scale kernel principal component analysis and its application for fault detection. *Chem. Eng. Res. Des.* **2012**, *90*, 1271–1280. [[CrossRef](#)]
135. Zhang, Y.; Li, S.; Hu, Z.; Song, C. Dynamical process monitoring using dynamical hierarchical kernel partial least squares. *Chemom. Intell. Lab. Syst.* **2012**, *118*, 150–158. [[CrossRef](#)]
136. 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](#)]
137. Guo, K.; San, Y.; Zhu, Y. Nonlinear process monitoring using wavelet kernel principal component analysis. In Proceedings of the 2012 International Conference on Systems and Informatics (ICSAI2012), Yantai, China, 19–20 May 2012; pp. 432–438. [[CrossRef](#)]
138. Sumana, C.; Detroja, K.; Gudi, R.D. Evaluation of nonlinear scaling and transformation for nonlinear process fault detection. *Int. J. Adv. Eng. Sci. Appl. Math.* **2012**, *4*, 52–66. [[CrossRef](#)]

139. Wang, Y.J.; Jia, M.X.; Mao, Z.Z. Weak fault monitoring method for batch process based on multi-model SDKPCA. *Chemom. Intell. Lab. Syst.* **2012**, *118*, 1–12. [[CrossRef](#)]
140. Liu, Y.; Wang, F.; Chang, Y. Reconstruction in integrating fault spaces for fault identification with kernel independent component analysis. *Chem. Eng. Res. Des.* **2013**, *91*, 1071–1084. [[CrossRef](#)]
141. Peng, K.; Zhang, K.; Li, G. Quality-related process monitoring based on total kernel PLS model and its industrial application. *Math. Probl. Eng.* **2013**, *2013*. [[CrossRef](#)]
142. Wang, Y.; Mao, Z.; Jia, M. Feature-points-based multimodel single dynamic kernel principle component analysis (M-SDKPCA) modeling and online monitoring strategy for uneven-length batch processes. *Ind. Eng. Chem. Res.* **2013**, *52*, 12059–12071. [[CrossRef](#)]
143. Jiang, Q.; Yan, X. Weighted kernel principal component analysis based on probability density estimation and moving window and its application in nonlinear chemical process monitoring. *Chemom. Intell. Lab. Syst.* **2013**, *127*, 121–131. [[CrossRef](#)]
144. Jiang, Q.; Yan, X. Statistical Monitoring of Chemical Processes Based on Sensitive Kernel Principal Components. *Chin. J. Chem. Eng.* **2013**, *21*, 633–643. [[CrossRef](#)]
145. Zhang, Y.; An, J.; Zhang, H. Monitoring of time-varying processes using kernel independent component analysis. *Chem. Eng. Sci.* **2013**, *88*, 23–32. [[CrossRef](#)]
146. Zhang, Y.; Zhang, L.; Lu, R. Fault identification of nonlinear processes. *Ind. Eng. Chem. Res.* **2013**, *52*, 12072–12081. [[CrossRef](#)]
147. Zhang, Y.; Wang, C.; Lu, R. Modeling and monitoring of multimode process based on subspace separation. *Chem. Eng. Res. Des.* **2013**, *91*, 831–842. [[CrossRef](#)]
148. Deng, X.; Tian, X. Nonlinear process fault pattern recognition using statistics kernel PCA similarity factor. *Neurocomputing* **2013**, *121*, 298–308. [[CrossRef](#)]
149. Deng, X.; Tian, X. Sparse kernel locality preserving projection and its application in nonlinear process fault detection. *Chin. J. Chem. Eng.* **2013**, *21*, 163–170. [[CrossRef](#)]
150. Deng, X.; Tian, X.; Chen, S. Modified kernel principal component analysis based on local structure analysis and its application to nonlinear process fault diagnosis. *Chemom. Intell. Lab. Syst.* **2013**, *127*, 195–209. [[CrossRef](#)]
151. Rong, G.; Liu, S.Y.; Shao, J.D. Fault diagnosis by Locality Preserving Discriminant Analysis and its kernel variation. *Comput. Chem. Eng.* **2013**, *49*, 105–113. [[CrossRef](#)]
152. Hu, Y.; Ma, H.; Shi, H. Enhanced batch process monitoring using just-in-time-learning based kernel partial least squares. *Chemom. Intell. Lab. Syst.* **2013**, *123*, 15–27. [[CrossRef](#)]
153. Hu, Y.; Ma, H.; Shi, H. Robust online monitoring based on spherical-kernel partial least squares for nonlinear processes with contaminated modeling data. *Ind. Eng. Chem. Res.* **2013**, *52*, 9155–9164. [[CrossRef](#)]
154. Fan, J.; Qin, S.J.; Wang, Y. Online monitoring of nonlinear multivariate industrial processes using filtering KICA-PCA. *Control Eng. Pract.* **2014**, *22*, 205–216. [[CrossRef](#)]
155. Zhang, Y.; Yang, N.; Li, S. Fault isolation of nonlinear processes based on fault directions and features. *IEEE Trans. Control Syst. Technol.* **2014**, *22*, 1567–1572. [[CrossRef](#)]
156. Zhang, Y.; Li, S. Modeling and monitoring of nonlinear multi-mode processes. *Control Eng. Pract.* **2014**, *22*, 194–204. [[CrossRef](#)]
157. Cai, L.; Tian, X.; Zhang, N. A kernel time structure independent component analysis method for nonlinear process monitoring. *Chin. J. Chem. Eng.* **2014**, *22*, 1243–1253. [[CrossRef](#)]
158. Wang, L.; Shi, H. Improved kernel PLS-based fault detection approach for nonlinear chemical processes. *Chin. J. Chem. Eng.* **2014**, *22*, 657–663. [[CrossRef](#)]
159. Elshenawy, L.M.; Mohamed, T.A.M. Fault Detection of Nonlinear Processes Using Fuzzy C-means-based Kernel PCA. In Proceedings of the International Conference on Machine Learning, Electrical and Mechanical Engineering (ICMLEME 2014), Dubai, UAE, 8–9 January 2014; International Institute of Engineers: Dubai, UAE, 2014. [[CrossRef](#)]
160. 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](#)]
161. Castillo, I.; Edgar, T.F.; Dunia, R. Nonlinear detection and isolation of multiple faults using residuals modeling. *Ind. Eng. Chem. Res.* **2014**, *53*, 5217–5233. [[CrossRef](#)]
162. Peng, K.X.; Zhang, K.; Li, G. Online Contribution Rate Based Fault Diagnosis for Nonlinear Industrial Processes. *Acta Autom. Sin.* **2014**, *40*, 423–430. [[CrossRef](#)]

163. Zhao, X.; Xue, Y. Output-relevant fault detection and identification of chemical process based on hybrid kernel T-PLS. *Can. J. Chem. Eng.* **2014**, *92*, 1822–1828. [[CrossRef](#)]
164. Godoy, J.L.; Zumoffen, D.A.; Vega, J.R.; Marchetti, J.L. New contributions to non-linear process monitoring through kernel partial least squares. *Chemom. Intell. Lab. Syst.* **2014**, *135*, 76–89. [[CrossRef](#)]
165. Kallas, M.; Mourrot, G.; Maquin, D.; Ragot, J. Diagnosis of nonlinear systems using kernel principal component analysis. *J. Phys. Conf. Ser.* **2014**, *570*. [[CrossRef](#)]
166. Ciabattini, L.; Comodi, G.; Ferracuti, F.; Fonti, A.; Giantomassi, A.; Longhi, S. Multi-apartment residential microgrid monitoring system based on kernel canonical variate analysis. *Neurocomputing* **2015**, *170*, 306–317. [[CrossRef](#)]
167. Li, N.; Yang, Y. Ensemble Kernel Principal Component Analysis for Improved Nonlinear Process Monitoring. *Ind. Eng. Chem. Res.* **2015**, *54*, 318–329. [[CrossRef](#)]
168. Liu, Y.; Zhang, G. Scale-sifting multiscale nonlinear process quality monitoring and fault detection. *Can. J. Chem. Eng.* **2015**, *93*, 1416–1425. [[CrossRef](#)]
169. Md Nor, N.; Hussain, M.A.; Hassan, C.R.C. Process Monitoring and Fault Detection in Non-Linear Chemical Process Based On Multi-Scale Kernel Fisher Discriminant Analysis. *Comput. Aided Chem. Eng.* **2015**, *37*, 1823–1828. [[CrossRef](#)]
170. Yao, M.; Wang, H. On-line monitoring of batch processes using generalized additive kernel principal component analysis. *J. Process Control* **2015**, *28*, 56–72. [[CrossRef](#)]
171. Wang, H.; Yao, M. Fault detection of batch processes based on multivariate functional kernel principal component analysis. *Chemom. Intell. Lab. Syst.* **2015**, *149*, 78–89. [[CrossRef](#)]
172. Huang, L.; Cao, Y.; Tian, X.; Deng, X. A Nonlinear Quality-relevant Process Monitoring Method with Kernel Input-output Canonical Variate Analysis. *IFAC-PapersOnLine* **2015**, *48*, 611–616. [[CrossRef](#)]
173. Zhang, Y.; Du, W.; Fan, Y.; Zhang, L. Process fault detection using directional kernel partial least squares. *Ind. Eng. Chem. Res.* **2015**, *54*, 2509–2518. [[CrossRef](#)]
174. Zhang, N.; Tian, X.; Cai, L.; Deng, X. Process fault detection based on dynamic kernel slow feature analysis. *Comput. Electr. Eng.* **2015**, *41*, 9–17. [[CrossRef](#)]
175. Zhang, H.; Tian, X.; Cai, L. Nonlinear Process Fault Diagnosis Using Kernel Slow Feature Discriminant Analysis. *IFAC-PapersOnLine* **2015**, *48*, 607–612. [[CrossRef](#)]
176. Zhang, Y.; Sun, R.; Fan, Y. Fault diagnosis of nonlinear process based on KCPLS reconstruction. *Chemom. Intell. Lab. Syst.* **2015**, *140*, 49–60. [[CrossRef](#)]
177. Samuel, R.T.; Cao, Y. Kernel canonical variate analysis for nonlinear dynamic process monitoring. *IFAC-PapersOnLine* **2015**, *28*, 605–610. [[CrossRef](#)]
178. Samuel, R.T.; Cao, Y. Improved kernel canonical variate analysis for process monitoring. In Proceedings of the 2015 21st International Conference on Automation and Computing (ICAC), Glasgow, UK, 11–12 September 2015; pp. 1–6. [[CrossRef](#)]
179. Chakour, C.; Harkat, M.F.; Djeghaba, M. New adaptive kernel principal component analysis for nonlinear dynamic process monitoring. *Appl. Math. Inf. Sci.* **2015**, *9*, 1833–1845.
180. Jiang, Q.; Yan, X. Nonlinear plant-wide process monitoring using MI-spectral clustering and Bayesian inference-based multiblock KPCA. *J. Process Control* **2015**, *32*, 38–50. [[CrossRef](#)]
181. Cai, E.; Liu, D.; Liang, L.; Xu, G. Monitoring of chemical industrial processes using integrated complex network theory with PCA. *Chemom. Intell. Lab. Syst.* **2015**, *140*, 22–35. [[CrossRef](#)]
182. Luo, L.; Bao, S.; Mao, J.; Tang, D. Nonlinear Process Monitoring Using Data-Dependent Kernel Global-Local Preserving Projections. *Ind. Eng. Chem. Res.* **2015**, *54*, 11126–11138. [[CrossRef](#)]
183. Bernal De Lázaro, J.M.; Prieto Moreno, A.; Llanes Santiago, O.; Da Silva Neto, A.J. Optimizing kernel methods to reduce dimensionality in fault diagnosis of industrial systems. *Comput. Ind. Eng.* **2015**, *87*, 140–149. [[CrossRef](#)]
184. Bernal-de Lázaro, J.M.; Llanes-Santiago, O.; Prieto-Moreno, A.; Knupp, D.C.; Silva-Neto, A.J. Enhanced dynamic approach to improve the detection of small-magnitude faults. *Chem. Eng. Sci.* **2016**, *146*, 166–179. [[CrossRef](#)]
185. Ji, H.; He, X.; Li, G.; Zhou, D. Determining the optimal kernel parameter in KPCA based on sample reconstruction. *Chin. Control Conf.* **2016**, *2016*, 6408–6414. [[CrossRef](#)]
186. Xu, Y.; Liu, Y.; Zhu, Q. Multivariate time delay analysis based local KPCA fault prognosis approach for nonlinear processes. *Chin. J. Chem. Eng.* **2016**, *24*, 1413–1422. [[CrossRef](#)]



187. Luo, L.; Bao, S.; Mao, J.; Tang, D. Nonlinear process monitoring based on kernel global-local preserving projections. *J. Process Control* **2016**, *38*, 11–21. [[CrossRef](#)]
188. Zhang, Y.; Fan, Y.; Yang, N. Fault diagnosis of multimode processes based on similarities. *IEEE Trans. Ind. Electron.* **2016**, *63*, 2606–2614. [[CrossRef](#)]
189. Taouali, O.; Jaffel, I.; Lahdhiri, H.; Harkat, M.F.; Messaoud, H. New fault detection method based on reduced kernel principal component analysis (RKPCA). *Int. J. Adv. Manuf. Technol.* **2016**, *85*, 1547–1552. [[CrossRef](#)]
190. Fazai, R.; Taouali, O.; Harkat, M.F.; Bouguila, N. A new fault detection method for nonlinear process monitoring. *Int. J. Adv. Manuf. Technol.* **2016**, *87*, 3425–3436. [[CrossRef](#)]
191. Jaffel, I.; Taouali, O.; Harkat, M.F.; Messaoud, H. Moving window KPCA with reduced complexity for nonlinear dynamic process monitoring. *ISA Trans.* **2016**, *64*, 184–192. [[CrossRef](#)]
192. Mansouri, M.; Nounou, M.; Nounou, H.; Karim, N. Kernel PCA-based GLRT for nonlinear fault detection of chemical processes. *J. Loss Prev. Process Ind.* **2016**, *40*, 334–347. [[CrossRef](#)]
193. Botre, C.; Mansouri, M.; Nounou, M.; Nounou, H.; Karim, M.N. Kernel PLS-based GLRT method for fault detection of chemical processes. *J. Loss Prev. Process Ind.* **2016**, *43*, 212–224. [[CrossRef](#)]
194. Samuel, R.T.; Cao, Y. Nonlinear process fault detection and identification using kernel PCA and kernel density estimation. *Syst. Sci. Control Eng.* **2016**, *4*, 165–174. [[CrossRef](#)]
195. 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](#)]
196. Jia, Q.; Du, W.; Zhang, Y. Semi-supervised kernel partial least squares fault detection and identification approach with application to HGPWLTP. *J. Chemom.* **2016**, *30*, 377–385. [[CrossRef](#)]
197. Jia, Q.; Zhang, Y. Quality-related fault detection approach based on dynamic kernel partial least squares. *Chem. Eng. Res. Des.* **2016**, *106*, 242–252. [[CrossRef](#)]
198. Jiang, Q.; Li, J.; Yan, X. Performance-driven optimal design of distributed monitoring for large-scale nonlinear processes. *Chemom. Intell. Lab. Syst.* **2016**, *155*, 151–159. [[CrossRef](#)]
199. Peng, K.; Zhang, K.; You, B.; Dong, J.; Wang, Z. A quality-based nonlinear fault diagnosis framework focusing on industrial multimode batch processes. *IEEE Trans. Ind. Electron.* **2016**, *63*, 2615–2624. [[CrossRef](#)]
200. Xie, L.; Li, Z.; Zeng, J.; Kruger, U. Block adaptive kernel principal component analysis for nonlinear process monitoring. *AIChE J.* **2016**, *62*, 4334–4345. [[CrossRef](#)]
201. Wang, G.; Luo, H.; Peng, K. Quality-related fault detection using linear and nonlinear principal component regression. *J. Franklin Inst.* **2016**, *353*, 2159–2177. [[CrossRef](#)]
202. Huang, J.; Yan, X. Related and independent variable fault detection based on KPCA and SVDD. *J. Process Control* **2016**, *39*, 88–99. [[CrossRef](#)]
203. Xiao, Y.-W.; Zhang, X.-H. Novel Nonlinear Process Monitoring and Fault Diagnosis Method Based on KPCA-ICA and MSVMs. *J. Control Autom. Electr. Syst.* **2016**, *27*, 289–299. [[CrossRef](#)]
204. Feng, J.; Wang, J.; Zhang, H.; Han, Z. Fault diagnosis method of joint fisher discriminant analysis based on the local and global manifold learning and its kernel version. *IEEE Trans. Autom. Sci. Eng.* **2016**, *13*, 122–133. [[CrossRef](#)]
205. Sheng, N.; Liu, Q.; Qin, S.J.; Chai, T. Comprehensive Monitoring of Nonlinear Processes Based on Concurrent Kernel Projection to Latent Structures. *IEEE Trans. Autom. Sci. Eng.* **2016**, *13*, 1129–1137. [[CrossRef](#)]
206. Zhang, Y.; Fan, Y.; Du, W. Nonlinear Process Monitoring Using Regression and Reconstruction Method. *IEEE Trans. Autom. Sci. Eng.* **2016**, *13*, 1343–1354. [[CrossRef](#)]
207. Jaffel, I.; Taouali, O.; Harkat, M.F.; Messaoud, H. Kernel principal component analysis with reduced complexity for nonlinear dynamic process monitoring. *Int. J. Adv. Manuf. Technol.* **2017**, *88*, 3265–3279. [[CrossRef](#)]
208. Lahdhiri, H.; Elaissi, I.; Taouali, O.; Harkat, M.F.; Messaoud, H. Nonlinear process monitoring based on new reduced Rank-KPCA method. *Stoch. Environ. Res. Risk Assess.* **2017**, *32*, 1833–1848. [[CrossRef](#)]
209. Lahdhiri, H.; Taouali, O.; Elaissi, I.; Jaffel, I.; Harkat, M.F.; Messaoud, H. A new fault detection index based on Mahalanobis distance and kernel method. *Int. J. Adv. Manuf. Technol.* **2017**, *91*, 2799–2809. [[CrossRef](#)]
210. Mansouri, M.; Nounou, M.N.; Nounou, H.N. Multiscale Kernel PLS-Based Exponentially Weighted-GLRT and Its Application to Fault Detection. *IEEE Trans. Emerg. Top. Comput. Intell.* **2017**, *3*, 49–58. [[CrossRef](#)]
211. Mansouri, M.; Nounou, M.N.; Nounou, H.N. Improved Statistical Fault Detection Technique and Application to Biological Phenomena Modeled by S-Systems. *IEEE Trans. Nanobiosci.* **2017**, *16*, 504–512. [[CrossRef](#)]

212. Sheriff, M.Z.; Karim, M.N.; Nounou, M.N.; Nounou, H.; Mansouri, M. Monitoring of chemical processes using improved multiscale KPCA. In Proceedings of the 2017 4th International Conference on Control, Decision and Information Technologies (CoDIT), Barcelona, Spain, 5–7 April 2017; pp. 49–54. [[CrossRef](#)]
213. Cai, L.; Tian, X.; Chen, S. Monitoring nonlinear and non-Gaussian processes using Gaussian mixture model-based weighted kernel independent component analysis. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 122–135. [[CrossRef](#)] [[PubMed](#)]
214. Zhang, H.; Qi, Y.; Wang, L.; Gao, X.; Wang, X. Fault detection and diagnosis of chemical process using enhanced KECA. *Chemom. Intell. Lab. Syst.* **2017**, *161*, 61–69. [[CrossRef](#)]
215. Zhang, H.; Tian, X.; Deng, X. Batch Process Monitoring Based on Multiway Global Preserving Kernel Slow Feature Analysis. *IEEE Access* **2017**, *5*, 2696–2710. [[CrossRef](#)]
216. Zhang, H.; Tian, X. Batch process monitoring based on batch dynamic Kernel slow feature analysis. In Proceedings of the 2017 29th Chinese Control And Decision Conference (CCDC), Chongqing, China, 28–30 May 2017; pp. 4772–4777. [[CrossRef](#)]
217. Zhang, Y.; Du, W.; Fan, Y.; Li, X. Comprehensive Correlation Analysis of Industrial Process. *IEEE Trans. Ind. Electron.* **2017**, *64*, 9461–9468. [[CrossRef](#)]
218. Zhang, Y.; Fu, Y.; Wang, Z.; Feng, L. Fault Detection Based on Modified Kernel Semi-Supervised Locally Linear Embedding. *IEEE Access* **2017**, *6*, 479–487. [[CrossRef](#)]
219. 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*, 1–12. [[CrossRef](#)]
220. Deng, X.; Tian, X.; Chen, S.; Harris, C.J. Deep learning based nonlinear principal component analysis for industrial process fault detection. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 1237–1243. [[CrossRef](#)]
221. Deng, X.; Zhong, N.; Wang, L. Nonlinear Multimode Industrial Process Fault Detection Using Modified Kernel Principal Component Analysis. *IEEE Access* **2017**, *5*, 23121–23132. [[CrossRef](#)]
222. Deng, X.; Tian, X.; Chen, S.; Harris, C.J. Fault discriminant enhanced kernel principal component analysis incorporating prior fault information for monitoring nonlinear processes. *Chemom. Intell. Lab. Syst.* **2017**, *162*, 21–34. [[CrossRef](#)]
223. Tan, R.; Samuel, R.T.; Cao, Y. Nonlinear Dynamic Process Monitoring: The Case Study of a Multiphase Flow Facility. *Comput. Aided Chem. Eng.* **2017**, *40*, 1495–1500. [[CrossRef](#)]
224. Shang, L.; Liu, J.; Zhang, Y. Efficient recursive kernel canonical variate analysis for monitoring nonlinear time-varying processes. *Can. J. Chem. Eng.* **2017**, *96*, 205–214. [[CrossRef](#)]
225. Li, G.; Peng, K.; Yuan, T.; Zhong, M. Kernel dynamic latent variable model for process monitoring with application to hot strip mill process. *Chemom. Intell. Lab. Syst.* **2017**, *171*, 218–225. [[CrossRef](#)]
226. Wang, G.; Jiao, J. A Kernel Least Squares Based Approach for Nonlinear Quality-Related Fault Detection. *IEEE Trans. Ind. Electron.* **2017**, *64*, 3195–3204. [[CrossRef](#)]
227. Wang, G.; Jiao, J.; Yin, S. A kernel direct decomposition-based monitoring approach for nonlinear quality-related fault detection. *IEEE Trans. Ind. Inform.* **2017**, *13*, 1565–1574. [[CrossRef](#)]
228. Wang, R.; Wang, J.; Zhou, J.; Wu, H. An improved kernel exponential discriminant analysis for fault identification of batch process. In Proceedings of the 2017 6th Data Driven Control and Learning Systems (DDCLS), Chongqing, China, 26–27 May 2017; pp. 16–21. [[CrossRef](#)]
229. Jiao, J.; Zhao, N.; Wang, G.; Yin, S. A nonlinear quality-related fault detection approach based on modified kernel partial least squares. *ISA Trans.* **2017**, *66*, 275–283. [[CrossRef](#)]
230. Huang, J.; Yan, X. Quality Relevant and Independent Two Block Monitoring Based on Mutual Information and KPCA. *IEEE Trans. Ind. Electron.* **2017**, *64*, 6518–6527. [[CrossRef](#)]
231. Yi, J.; Huang, D.; He, H.; Zhou, W.; Han, Q.; Li, T. A novel framework for fault diagnosis using kernel partial least squares based on an optimal preference matrix. *IEEE Trans. Ind. Electron.* **2017**, *64*, 4315–4324. [[CrossRef](#)]
232. Md Nor, N.; Hussain, M.A.; Che Hassan, C.R. Fault diagnosis and classification framework using multi-scale classification based on kernel Fisher discriminant analysis for chemical process system. *Appl. Soft Comput.* **2017**, *61*, 959–972. [[CrossRef](#)]
233. Du, W.; Fan, Y.; Zhang, Y.; Zhang, J. Fault diagnosis of non-Gaussian process based on FKICA. *J. Frankl. Inst.* **2017**, *354*, 2573–2590. [[CrossRef](#)]

234. Zhang, S.; Zhao, C. Stationarity test and Bayesian monitoring strategy for fault detection in nonlinear multimode processes. *Chemom. Intell. Lab. Syst.* **2017**, *168*, 45–61. [[CrossRef](#)]
235. Zhou, L.; Chen, J.; Yao, L.; Song, Z.; Hou, B. Similarity based robust probability latent variable regression model and its kernel extension for process monitoring. *Chemom. Intell. Lab. Syst.* **2017**, *161*, 88–95. [[CrossRef](#)]
236. Gharahbagheri, H.; Imtiaz, S.A.; Khan, F. Root Cause Diagnosis of Process Fault Using KPCA and Bayesian Network. *Ind. Eng. Chem. Res.* **2017**. [[CrossRef](#)]
237. Gharahbagheri, H.; Imtiaz, S.; Khan, F. Combination of KPCA and causality analysis for root cause diagnosis of industrial process fault. *Can. J. Chem. Eng.* **2017**, *95*, 1497–1509. [[CrossRef](#)]
238. Galiaskarov, M.R.; Kurkina, V.V.; Rusinov, L.A. Online diagnostics of time-varying nonlinear chemical processes using moving window kernel principal component analysis and Fisher discriminant analysis. *J. Chemom.* **2017**, e2866. [[CrossRef](#)]
239. Zhu, Q.X.; Meng, Q.Q.; He, Y.L. Novel Multidimensional Feature Pattern Classification Method and Its Application to Fault Diagnosis. *Ind. Eng. Chem. Res.* **2017**, *56*, 8906–8916. [[CrossRef](#)]
240. Zhu, Q.; Liu, Q.; Qin, S.J. Quality-relevant fault detection of nonlinear processes based on kernel concurrent canonical correlation analysis. *Proc. Am. Control Conf.* **2017**, pp. 5404–5409. [[CrossRef](#)]
241. Liu, Q.; Zhu, Q.; Qin, S.J.; Chai, T. Dynamic concurrent kernel CCA for strip-thickness relevant fault diagnosis of continuous annealing processes. *J. Process Control* **2018**, *67*, 12–22. [[CrossRef](#)]
242. Wang, G.; Jiao, J. Nonlinear Fault Detection Based on An Improved Kernel Approach. *IEEE Access* **2018**, *6*, 11017–11023. [[CrossRef](#)]
243. Wang, L. Enhanced fault detection for nonlinear processes using modified kernel partial least squares and the statistical local approach. *Can. J. Chem. Eng.* **2018**, *96*, 1116–1126. [[CrossRef](#)]
244. Huang, J.; Yan, X. Quality-Driven Principal Component Analysis Combined With Kernel Least Squares for Multivariate Statistical Process Monitoring. *IEEE Trans. Control Syst. Technol.* **2018**, *27*, 2688–2695. [[CrossRef](#)]
245. Huang, J.; Yan, X. Relevant and independent multi-block approach for plant-wide process and quality-related monitoring based on KPCA and SVDD. *ISA Trans.* **2018**, *73*, 257–267. [[CrossRef](#)]
246. Fezai, R.; Mansouri, M.; Taouali, O.; Harkat, M.F.; Bouguila, N. Online reduced kernel principal component analysis for process monitoring. *J. Process Control* **2018**, *61*, 1–11. [[CrossRef](#)]
247. Fezai, 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](#)]
248. Mansouri, M.; Baklouti, R.; Harkat, M.F.; Nounou, M.; Nounou, H.; Hamida, A.B. Kernel Generalized Likelihood Ratio Test for Fault Detection of Biological Systems. *IEEE Trans. Nanobiosci.* **2018**, *17*, 498–506. [[CrossRef](#)] [[PubMed](#)]
249. Jaffel, I.; Taouali, O.; Harkat, M.F.; Messaoud, H. Fault detection and isolation in nonlinear systems with partial Reduced Kernel Principal Component Analysis method. *Trans. Inst. Meas. Control* **2018**, *40*, 1289–1296. [[CrossRef](#)]
250. Lahdhiri, H.; Ben Abdellafou, K.; Taouali, O.; Mansouri, M.; Korbaa, O. New online kernel method with the Tabu search algorithm for process monitoring. *Trans. Inst. Meas. Control* **2018**. [[CrossRef](#)]
251. Tan, R.; Cao, Y. Deviation Contribution Plots of Multivariate Statistics. *IEEE Trans. Ind. Inform.* **2019**, *15*, 833–841. [[CrossRef](#)]
252. He, F.; Wang, C.; Fan, S.K.S. Nonlinear fault detection of batch processes based on functional kernel locality preserving projections. *Chemom. Intell. Lab. Syst.* **2018**, *183*, 79–89. [[CrossRef](#)]
253. Navi, M.; Meskin, N.; Davoodi, M. Sensor fault detection and isolation of an industrial gas turbine using partial adaptive KPCA. *J. Process Control* **2018**, *64*, 37–48. [[CrossRef](#)]
254. Chakour, C.; Benyounes, A.; Boudiaf, M. Diagnosis of uncertain nonlinear systems using interval kernel principal components analysis: Application to a weather station. *ISA Trans.* **2018**, *83*, 126–141. [[CrossRef](#)]
255. Deng, X.; Wang, L. Modified kernel principal component analysis using double-weighted local outlier factor and its application to nonlinear process monitoring. *ISA Trans.* **2018**, *72*, 218–228. [[CrossRef](#)]
256. Deng, X.; Tian, X.; Chen, S.; Harris, C.J. Deep Principal Component Analysis Based on Layerwise Feature Extraction and Its Application to Nonlinear Process Monitoring. *IEEE Trans. Control Syst. Technol.* **2018**, *27*, 2526–2540. [[CrossRef](#)]
257. Deng, X.; Tian, X.; Chen, S.; Harris, C.J. Nonlinear Process Fault Diagnosis Based on Serial Principal Component Analysis. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 560–572. [[CrossRef](#)] [[PubMed](#)]

258. Deng, X.; Sun, B.; Wang, L. Improved kernel fisher discriminant analysis for nonlinear process fault pattern recognition. In Proceedings of the 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS), Enshi, China, 25–27 May 2018; pp. 33–37. [\[CrossRef\]](#)
259. Zhang, H.; Tian, X.; Deng, X.; Cao, Y. Batch process fault detection and identification based on discriminant global preserving kernel slow feature analysis. *ISA Trans.* **2018**, *79*, 108–126. [\[CrossRef\]](#) [\[PubMed\]](#)
260. Shang, J.; Chen, M.; Zhang, H. Fault detection based on augmented kernel Mahalanobis distance for nonlinear dynamic processes. *Comput. Chem. Eng.* **2018**, *109*, 311–321. [\[CrossRef\]](#)
261. Jiang, Q.; Yan, X. Parallel PCA–KPCA for nonlinear process monitoring. *Control Eng. Pract.* **2018**, *80*, 17–25. [\[CrossRef\]](#)
262. Feng, L.; Di, T.; Zhang, Y. HSIC-based kernel independent component analysis for fault monitoring. *Chemom. Intell. Lab. Syst.* **2018**, *178*, 47–55. [\[CrossRef\]](#)
263. Zhao, C.; Huang, B. Incipient Fault Detection for Complex Industrial Processes with Stationary and Nonstationary Hybrid Characteristics. *Ind. Eng. Chem. Res.* **2018**, *57*. [\[CrossRef\]](#)
264. Zhai, L.; Zhang, Y.; Guan, S.; Fu, Y.; Feng, L. Nonlinear process monitoring using kernel nonnegative matrix factorization. *Can. J. Chem. Eng.* **2018**, *96*, 554–563. [\[CrossRef\]](#)
265. Ma, J.; Li, G.; Zhou, D. Fault prognosis technology for non-Gaussian and nonlinear processes based on KICA reconstruction. *Can. J. Chem. Eng.* **2018**, *96*, 515–520. [\[CrossRef\]](#)
266. Lu, Q.; Jiang, B.; Gopaluni, R.B.; Loewen, P.D.; Braatz, R.D. Locality preserving discriminative canonical variate analysis for fault diagnosis. *Comput. Chem. Eng.* **2018**, *117*, 309–319. [\[CrossRef\]](#)
267. Li, W.; Zhao, C.; Gao, F. Linearity Evaluation and Variable Subset Partition Based Hierarchical Process Modeling and Monitoring. *IEEE Trans. Ind. Electron.* **2018**, *65*, 2683–2692. [\[CrossRef\]](#)
268. Chu, F.; Dai, W.; Shen, J.; Ma, X.; Wang, F. Online complex nonlinear industrial process operating optimality assessment using modified robust total kernel partial M-regression. *Chin. J. Chem. Eng.* **2018**, *26*, 775–785. [\[CrossRef\]](#)
269. Zhai, L.; Jia, Q. Simultaneous fault detection and isolation using semi-supervised kernel nonnegative matrix factorization. *Can. J. Chem. Eng.* **2019**, 1–10. [\[CrossRef\]](#)
270. Fezai, R.; Mansouri, M.; Trabelsi, M.; Hajji, M.; Nounou, H.; Nounou, M. Online reduced kernel GLRT technique for improved fault detection in photovoltaic systems. *Energy* **2019**, *179*, 1133–1154. [\[CrossRef\]](#)
271. Fazai, R.; Mansouri, M.; Abodayeh, K.; Nounou, H.; Nounou, M. Online reduced kernel PLS combined with GLRT for fault detection in chemical systems. *Process Saf. Environ. Prot.* **2019**, *128*, 228–243. [\[CrossRef\]](#)
272. Deng, X.; Deng, J. Incipient Fault Detection for Chemical Processes Using Two-Dimensional Weighted SLKPCA. *Ind. Eng. Chem. Res.* **2019**, *58*, 2280–2295. [\[CrossRef\]](#)
273. Cui, P.; Zhan, C.; Yang, Y. Improved nonlinear process monitoring based on ensemble KPCA with local structure analysis. *Chem. Eng. Res. Des.* **2019**, *142*, 355–368. [\[CrossRef\]](#)
274. Lahdhiri, H.; Said, M.; Abdellafou, K.B.; Taouali, O.; Harkat, M.F. Supervised process monitoring and fault diagnosis based on machine learning methods. *Int. J. Adv. Manuf. Technol.* **2019**, *102*, 2321–2337. [\[CrossRef\]](#)
275. Liu, Y.; Wang, F.; Chang, Y.; Gao, F.; He, D. Performance-relevant kernel independent component analysis based operating performance assessment for nonlinear and non-Gaussian industrial processes. *Chem. Eng. Sci.* **2019**, *209*, 115167. [\[CrossRef\]](#)
276. Liu, M.; Li, X.; Lou, C.; Jiang, J. A fault detection method based on CPSO-improved KICA. *Entropy* **2019**, *21*, 668. [\[CrossRef\]](#)
277. Yu, J.; Wang, K.; Ye, L.; Song, Z. Accelerated Kernel Canonical Correlation Analysis with Fault Relevance for Nonlinear Process Fault Isolation. *Ind. Eng. Chem. Res.* **2019**, *58*, 18280–18291. [\[CrossRef\]](#)
278. Guo, L.; Wu, P.; Gao, J.; Lou, S. Sparse Kernel Principal Component Analysis via Sequential Approach for Nonlinear Process Monitoring. *IEEE Access* **2019**, *7*, 47550–47563. [\[CrossRef\]](#)
279. Wu, P.; Guo, L.; Lou, S.; Gao, J. Local and Global Randomized Principal Component Analysis for Nonlinear Process Monitoring. *IEEE Access* **2019**, *7*, 25547–25562. [\[CrossRef\]](#)
280. Harkat, M.F.; Mansouri, M.; Nounou, M.; Nounou, H. Fault detection of uncertain nonlinear process using interval-valued data-driven approach. *Chem. Eng. Sci.* **2019**, *205*, 36–45. [\[CrossRef\]](#)
281. Ma, L.; Dong, J.; Peng, K. A Novel Hierarchical Detection and Isolation Framework for Quality-Related Multiple Faults in Large-Scale Processes. *IEEE Trans. Ind. Electron.* **2019**, *67*, 1316–1327. [\[CrossRef\]](#)
282. Zhang, H.; Deng, X.; Zhang, Y.; Hou, C.; Li, C.; Xin, Z. Nonlinear Process Monitoring Based on Global Preserving Unsupervised Kernel Extreme Learning Machine. *IEEE Access* **2019**, *7*, 106053–106064. [\[CrossRef\]](#)

283. Peng, K.; Ren, Z.; Dong, J.; Ma, L. A New Hierarchical Framework for Detection and Isolation of Multiple Faults in Complex Industrial Processes. *IEEE Access* **2019**, *7*, 12006–12015. [[CrossRef](#)]
284. Yan, S.; Huang, J.; Yan, X. Monitoring of quality-relevant and quality-irrelevant blocks with characteristic-similar variables based on self-organizing map and kernel approaches. *J. Process Control* **2019**, *73*, 103–112. [[CrossRef](#)]
285. Huang, K.; Wen, H.; Ji, H.; Cen, L.; Chen, X.; Yang, C. Nonlinear process monitoring using kernel dictionary learning with application to aluminum electrolysis process. *Control Eng. Pract.* **2019**, *89*, 94–102. [[CrossRef](#)]
286. Zhou, Z.; Du, N.; Xu, J.; Li, Z.; Wang, P.; Zhang, J. Randomized Kernel Principal Component Analysis for Modeling and Monitoring of Nonlinear Industrial Processes with Massive Data. *Ind. Eng. Chem. Res.* **2019**, *58*, 10410–10417. [[CrossRef](#)]
287. Deng, J.; Deng, X.; Wang, L.; Zhang, X. Nonlinear Process Monitoring Based on Multi-block Dynamic Kernel Principal Component Analysis. In Proceedings of the 2018 13th World Congress on Intelligent Control and Automation (WCICA), Changsha China, 4–8 July 2018; pp. 1058–1063. [[CrossRef](#)]
288. Wang, G.; Jiao, J.; Yin, S. Efficient Nonlinear Fault Diagnosis Based on Kernel Sample Equivalent Replacement. *IEEE Trans. Ind. Inform.* **2019**, *15*, 2682–2690. [[CrossRef](#)]
289. Zhu, W.; Zhen, W.; Jiao, J. Partial Derivate Contribution Plot Based on KPLS-KSER for Nonlinear Process Fault Diagnosis. In Proceedings of the 34th Youth Academic Annual Conference of Chinese Association of Automation, Jinzhou, China, 6–8 June 2019; pp. 735–740. [[CrossRef](#)]
290. Xiao, S. Locality Kernel Canonical Variate Analysis for Fault Detection. *J. Phys. Conf. Ser.* **2019**, *1284*, 012003. [[CrossRef](#)]
291. Xiao, S. Kernel Canonical Variate Dissimilarity Analysis for Fault Detection. *Chin. Control Conf.* **2019**, *1284*, 6871–6876. [[CrossRef](#)]
292. Shang, L.; Yan, Z.; Qiu, A.; Li, F.; Zhou, X. Efficient recursive kernel principal component analysis for nonlinear time-varying processes monitoring. In Proceedings of the 2019 Chinese Control And Decision Conference (CCDC), Nanchang, China, 3–5 June 2019; pp. 3057–3062. [[CrossRef](#)]
293. Geng, Z.; Liu, F.; Han, Y.; Zhu, Q.; He, Y. Fault Diagnosis of Chemical Processes Based on a novel Adaptive Kernel Principal Component Analysis. In Proceedings of the 2019 12th Asian Control Conference (ASCC), Kitakyushu-shi, Japan, 9–12 June 2019; pp. 1495–1500.
294. Md Nor, N.; Hussain, M.A.; Che Hassan, C.R. Multi-scale kernel Fisher discriminant analysis with adaptive neuro-fuzzy inference system (ANFIS) in fault detection and diagnosis framework for chemical process systems. *Neural Comput. Appl.* **2019**, *9*. [[CrossRef](#)]
295. Tan, R.; Cong, T.; Thornhill, N.F.; Ottewill, J.R.; Baranowski, J. Statistical Monitoring of Processes with Multiple Operating Modes. *IFAC-PapersOnLine* **2019**, *52*, 635–642. [[CrossRef](#)]
296. Tan, R.; Ottewill, J.R.; Thornhill, N.F. Nonstationary Discrete Convolution Kernel for Multimodal Process Monitoring. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, 1–12. [[CrossRef](#)]
297. Yao, Y.; Gao, F. A survey on multistage/multiphase statistical modeling methods for batch processes. *Annu. Rev. Control* **2009**, *33*, 172–183. [[CrossRef](#)]
298. Rendall, R.; Chiang, L.H.; Reis, M.S. Data-driven methods for batch data analysis—A critical overview and mapping on the complexity scale. *Comput. Chem. Eng.* **2019**, *124*, 1–13. [[CrossRef](#)]
299. Larimore, W.E. Canonical variate analysis in identification, filtering, and adaptive control. In Proceedings of the 29th IEEE Conference on Decision and Control, Honolulu, HI, USA, 5–7 December 1990; Volume 2, pp. 596–604. [[CrossRef](#)]
300. Wiskott, L.; Sejnowski, T.J. Slow feature analysis: Unsupervised learning of invariances. *Neural Comput.* **2002**, *14*, 715–770. [[CrossRef](#)] [[PubMed](#)]
301. Jiang, B.; Huang, D.; Zhu, X.; Yang, F.; Braatz, R.D. Canonical variate analysis-based contributions for fault identification. *J. Process Control* **2015**, *26*, 17–25. [[CrossRef](#)]
302. Krzanowski, W.J. Between-Groups Comparison of Principal Components. *J. Am. Stat. Assoc.* **1979**, *74*, 703–707. [[CrossRef](#)]
303. Ge, Z.; Song, Z. Process monitoring based on independent Component Analysis-Principal Component Analysis (ICA-PCA) and similarity factors. *Ind. Eng. Chem. Res.* **2007**, *46*, 2054–2063. [[CrossRef](#)]
304. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297.10.1023/A:1022627411411. [[CrossRef](#)]

305. Zhang, Y. Fault Detection and Diagnosis of Nonlinear Processes Using Improved Kernel Independent Component Analysis (KICA) and Support Vector Machine (SVM). *Ind. Eng. Chem. Res.* **2008**, *47*, 6961–6971. [[CrossRef](#)]
306. Gönen, M.; Alpaydın, E. Multiple Kernel Learning Algorithms. *J. Mach. Learn. Res.* **2011**, *12*, 2211–2268.
307. Yu, J.; Rashid, M.M. A novel dynamic bayesian network-based networked process monitoring approach for fault detection, propagation identification, and root cause diagnosis. *AIChE J.* **2013**, *59*, 2348–2365. [[CrossRef](#)]
308. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning*; Springer: New York, NY, USA, 2008.
309. Hyvärinen, A.; Oja, E. Independent component analysis: Algorithms and applications. *Neural Netw.* **2000**, *13*, 411–430. [[CrossRef](#)]
310. Kano, M.; Nagao, K.; Ohno, H.; Hasebe, S.; Hashimoto, I. Dissimilarity of Process Data for Statistical Process Monitoring. *IFAC Proc. Vol.* **2000**, *33*, 231–236. [[CrossRef](#)]
311. Rashid, M.M.; Yu, J. A new dissimilarity method integrating multidimensional mutual information and independent component analysis for non-Gaussian dynamic process monitoring. *Chemom. Intell. Lab. Syst.* **2012**, *115*, 44–58. [[CrossRef](#)]
312. He, Q.P.; Wang, J. Statistics pattern analysis: A new process monitoring framework and its application to semiconductor batch processes. *AIChE J.* **2011**, *57*, 107–121. [[CrossRef](#)]
313. Genton, M.G. Classes of Kernels for Machine Learning: A Statistics Perspective. *J. Mach. Learn. Res.* **2001**, *2*, 299–312. [[CrossRef](#)]
314. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006. [[CrossRef](#)]
315. Pilario, K.E.S. 2019. Available online: <https://uk.mathworks.com/matlabcentral/fileexchange/69941-kernel-pca-contour-maps-for-fault-detection> (accessed on 25 April 2019).
316. Halim, S.; Halim, F. *Competitive Programming 3: The New Lower Bound of Programming Contests*; Lulu Press: Morrisville, NC, USA, 2013.
317. Baudat, G.; Anouar, F. Feature vector selection and projection using kernels. *Neurocomputing* **2003**, *55*, 21–38. [[CrossRef](#)]
318. Yang, T.; Li, Y.F.; Mahdavi, M.; Jin, R.; Zhou, Z.H. Nyström Method vs Random Fourier Features: A Theoretical and Empirical Comparison. *Adv. NIPS* **2012**, pp. 485–493.
319. Saul, L.K.; Roweis, S. Think Globally, Fit Locally: Unsupervised Learning of Low Dimensional Manifolds. *J. Mach. Learn. Res.* **2003**, *4*, 119–155. [[CrossRef](#)]
320. Van Der Maaten, L.J.P.; Postma, E.O.; Van Den Herik, H.J. Dimensionality Reduction: A Comparative Review. *J. Mach. Learn. Res.* **2009**, *10*, 1–41. [[CrossRef](#)]
321. He, X.; Niyogi, P. Locality Preserving Projections. In Proceedings of the 16th International Conference Neural Information Processing Systems, Whistler, BC, Canada; 9–11 December 2003; pp. 153–160.
322. Hu, K.; Yuan, J. Multivariate statistical process control based on multiway locality preserving projections. *J. Process Control* **2008**, *18*, 797–807. [[CrossRef](#)]
323. Ham, J.; Lee, D.D.; Mika, S.; Schölkopf, B. A kernel view of the dimensionality reduction of manifolds. In Proceedings of the 21st International Machine Learning Conference (ICML '04), Banff, AB, Canada, 4–8 July 2004; Volume 12, p. 47. [[CrossRef](#)]
324. Hoegaerts, L.; De Lathauwer, L.; Goethals, I.; Suykens, J.A.K.; Vandewalle, J.; De Moor, B. Efficiently updating and tracking the dominant kernel principal components. *Neural Netw.* **2007**, *20*, 220–229. [[CrossRef](#)]
325. Hall, P.; Marshall, D.; Martin, R. Adding and subtracting eigenspaces with eigenvalue decomposition and singular value decomposition. *Image Vis. Comput.* **2002**, *20*, 1009–1016. [[CrossRef](#)]
326. Jiang, Q.; Huang, B. Distributed monitoring for large-scale processes based on multivariate statistical analysis and Bayesian method. *J. Process Control* **2016**, *46*, 75–83. [[CrossRef](#)]
327. Jiang, Q.; Yan, X. Plant-wide process monitoring based on mutual information-multiblock principal component analysis. *ISA Trans.* **2014**, *53*, 1516–1527. [[CrossRef](#)]
328. Melis, G. Dissecting the Winning Solution of the HiggsML Challenge. *J. Mach. Learn. Res. Work. Conf. Proc.* **2015**, *42*, 57–67.
329. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2006; ACM Press: New York, NY, USA, 2016; Volume 19, pp. 785–794. [[CrossRef](#)]

330. Vinyals, O.; Babuschkin, I.; Czarnecki, W.M.; Mathieu, M.; Dudzik, A.; Chung, J.; Choi, D.H.; Powell, R.; Ewalds, T.; Georgiev, P.; et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* **2019**, *575*, 350–354. [[CrossRef](#)] [[PubMed](#)]
331. Lucke, M.; Stief, A.; Chioua, M.; Ottewill, J.R.; Thornhill, N.F. Fault detection and identification combining process measurements and statistical alarms. *Control Eng. Pract.* **2020**, *94*, 104195. [[CrossRef](#)]
332. Ruiz-Cárcel, C.; Jaramillo, V.H.; Mba, D.; Ottewill, J.R.; Cao, Y. Combination of process and vibration data for improved condition monitoring of industrial systems working under variable operating conditions. *Mech. Syst. Signal Process.* **2015**, *66–67*, 699–714. [[CrossRef](#)]
333. Stief, A.; Tan, R.; Cao, Y.; Ottewill, J.R.; Thornhill, N.F.; Baranowski, J. A heterogeneous benchmark dataset for data analytics: Multiphase flow facility case study. *J. Process Control* **2019**, *79*, 41–55. [[CrossRef](#)]
334. Vachtsevanos, G.; Lewis, F.L.; Roemer, M.; Hess, A.; Wu, B. *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2006.
335. Ge, Z. Distributed predictive modeling framework for prediction and diagnosis of key performance index in plant-wide processes. *J. Process Control* **2018**, *65*, 107–117. [[CrossRef](#)]



© 2019 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 (<http://creativecommons.org/licenses/by/4.0/>).