

Article

Investigating Machine Learning and Control Theory Approaches for Process Fault Detection: A Comparative Study of KPCA and the Observer-Based Method

Fatma Lajmi ^{1,*}, Lotfi Mhamdi ², Wiem Abdelbaki ^{3,*}, Hedi Dhouibi ⁴ and Khaled Younes ³

¹ National Engineering School of Sousse, ENISO Laboratory: Networked Objects, Control, and Communication Systems (NOCCS), Sousse 4054, Tunisia

² National School of Engineering Monastir, Rue Ibn ELJazzar, Monastir 5019, Tunisia; lotfienim@yahoo.fr

³ College of Engineering and Technology, American University of the Middle East, Egaila 54200, Kuwait; khaled.younes@aum.edu.kw

⁴ High Institute of Applied Sciences and Technology of Kairouan, University of Kairouan, Kairouan 3100, Tunisia; hedi.dhouibi@laposte.net

* Correspondence: flajmi@yahoo.fr (F.L.); wiem.abdelbaki@aum.edu.kw (W.A.)

Abstract: The paper focuses on the importance of prompt and efficient process fault detection in contemporary manufacturing industries, where product quality and safety protocols are critical. The study compares the efficiencies of two techniques for process fault detection: Kernel Principal Component Analysis (KPCA) and the observer method. Both techniques are applied to observe water volume variation within a hydraulic system comprising three tanks. PCA is an unsupervised learning technique used for dimensionality reduction and pattern recognition. It is an extension of Principal Component Analysis (PCA) that utilizes kernel functions to transform data into higher-dimensional spaces, where it becomes easier to separate classes or identify patterns. In this paper, KPCA is applied to detect faults in the hydraulic system by analyzing the variation in water volume. The observer method originates from control theory and is utilized to estimate the internal states of a system based on its output measurements. It is commonly used in control systems to estimate the unmeasurable or hidden states of a system, which is crucial for ensuring proper control and fault detection. In this study, the observer method is applied to the hydraulic system to estimate the water volume variations within the three tanks. The paper presents a comparative study of these two techniques applied to the hydraulic system. The results show that both KPCA and the observer method perform similarly in detecting faults within the system. This similarity in performance highlights the efficacy of these techniques and their potential adaptability in various fault diagnosis scenarios within modern manufacturing processes.

Keywords: KPCA; observer-based method; machine learning; unsupervised learning; nonlinear process monitoring; fault detection



Citation: Lajmi, F.; Mhamdi, L.; Abdelbaki, W.; Dhouibi, H.; Younes, K. Investigating Machine Learning and Control Theory Approaches for Process Fault Detection: A Comparative Study of KPCA and the Observer-Based Method. *Sensors* **2023**, *23*, 6899. <https://doi.org/10.3390/s23156899>

Academic Editor: Kim Phuc Tran

Received: 26 June 2023

Revised: 25 July 2023

Accepted: 1 August 2023

Published: 3 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The contemporary manufacturing landscape necessitates heightened product quality and safety operational practices. To maintain optimal system functionality and reduce downtime in case of failure, the early detection of process faults is critical [1]. Consequently, several process monitoring-based Multivariate Statistical Process (MSP) methods were developed thanks to their efficiencies and simplicity [2,3].

The Kernel Principal Component Analysis (KPCA) method, a simple yet interesting technique developed by Ratch et al. [4], is designed to accurately model nonlinear relationships inherent in process data. Utilizing the principle of kernel tricks [5], KPCA can efficiently project the input data with linearly inseparable structures onto a higher dimensional feature space in which the data become linearly separable, facilitating the

execution of conventional Principal Component Analysis (PCA) within the feature space. KPCA has shown good results in analysis, modeling, and fault detection accuracy across a range of applications such as face recognition [6], speech recognition, nonlinear process monitoring, and fault diagnosis [7,8].

The observer-based method is commonly employed in Fault Detection (FD). The fundamental concept behind the observer or filter-based technique involves approximating the system's states using the collected measurement data. Consequently, the estimated states are compared to the measured states of the monitored system to produce the residual.

Different attempts have addressed the incorporation of the observer in control systems, specifically in the domains of electrical drives [9,10] and robotics [11]. The primary objective when designing the observer for control applications should focus on accurate state estimation.

The estimation of unmeasured states differs from the design of observers for fault detection, which focuses on estimating measured states [12]. Numerous studies have explored observer-based FD in various contexts [13–15].

For instance, Khalid et al. [16] detected sensor faults in small autonomous helicopters using Observer/Kalman filter identification. Yang and Liu [17] employed a Kalman filter for fault diagnosis and a hybrid genetic adaptive neuro-fuzzy inference system for fault classification.

Alkaya and Eker [18] introduced a state estimation approach using Kalman filtering to predict failures and prevent maintenance in a DC motor. Tarantino et al. [19] detected sensor faults in DC and BLDC motors using Luenberger observers.

Several issues are emerging, mainly related to fault diagnosis. The majority of them investigate single faults. However, a more intricate issue demands attention, namely the detection and localization of multiple faults—a situation where multiple breakdowns occur and overlap in time. We assess the system using a fault injection approach, where we deliberately introduce faults and observe system responses while adjusting the time parameter each time.

This problem occurs in many industrial systems. Nonetheless, the research dedicated to this domain remains limited due to the complexity of the task and the combinatorial explosion problem. In fact, the complexity of the current systems, with their multiple functionalities, makes them susceptible to multiple faults, which need to be considered simultaneously. The heterogeneity of the data that should be explored for diagnosis poses an additional challenge and requires adapted methods. The purpose of this paper is to propose a method for fault diagnosis in complex physical systems. Techniques derived from machine learning and control theory are used for this purpose, KPCA [20] and the observer method [21,22], respectively. We used KPCA and observer methods to simulate and detect faults in the three-tank system.

The rest of the article is structured as follows: Section 2 presents KPCA and its fault detection index. Section 3 introduces the observer approach and its underlying principles. In Section 4, a comparative study between the observer approach and the kernel method for fault detection in a hydraulic system with three tanks is presented. Conclusions are delivered at the conclusion of the paper.

2. Kernel Principal Component Analysis (KPCA)

Kernel Principal Component Analysis (KPCA) is a nonlinear extension of the traditional Principal Component Analysis (PCA) technique. PCA is a widely used method for dimensionality reduction and feature extraction in data analysis. However, PCA assumes linearity in the data, which limits its applicability to linearly separable or well-behaved datasets.

KPCA overcomes this limitation by employing a kernel function to map the input data into a higher-dimensional feature space, where it becomes easier to find nonlinear relationships and patterns. In this feature space, KPCA applies PCA to extract the principal components, which are the directions of maximum variance in the data.

The kernel function in KPCA allows for implicit computations in the high-dimensional space without explicitly transforming the data. Commonly used kernel functions include the Gaussian (or radial basis function), polynomial, and sigmoid kernels. These kernels measure the similarity or dissimilarity between data points, enabling the extraction of nonlinear features that would be challenging to capture using linear techniques.

The KPCA algorithm involves three main steps: (1) computation of the kernel matrix, which stores the pairwise similarities between data points based on the chosen kernel function, (2) agenda composition of the kernel matrix to obtain the principal components, and (3) projection of the data onto the principal components to obtain the transformed features.

KPCA finds applications in various fields, including computer vision, pattern recognition, bioinformatics, and signal processing. It enables the detection of nonlinear patterns, clustering of complex data, and nonlinear dimensionality reduction. By leveraging the power of nonlinear mapping, KPCA offers a valuable tool for analyzing and extracting meaningful information from high-dimensional and nonlinear datasets for education and analysis [23,24]. It consists of transforming the nonlinear aspects of input data space E into linear ones within a newly high-dimensional feature space, denoted H , and to perform PCA in that space. The feature space H is nonlinearly transformed from the input space E with a non-linear mapping function ϕ . The mapping of sample $x \in E$ in the feature space H can be written as:

$$\begin{aligned} \phi : E \subset \mathbb{R}^m &\rightarrow H \subset \mathbb{R}^H \\ x &\rightarrow \phi(x) \end{aligned} \quad (1)$$

Let us consider $X = [x_1, \dots, x_i, \dots, x_N]^T$ the training data matrix scaled to zero mean and unit variance. Where $x_i \in E \subset \mathbb{R}^m$ is a data vector, N is the number of observation samples and m is the number of process variables.

The monitoring phase based on the linear PCA approach requires the selection of principal components that maximize the variance in the data set. This is accomplished using the eigen decomposition of the covariance matrix. Similarly, this approach was generalized in the Kernel PCA approach by Ratsch [4]. The covariance matrix C_ϕ in the feature space H is given by:

$$C_\phi = \frac{1}{N} \sum_{i=1}^N \phi(x_i) \phi(x_i)^T \quad (2)$$

Let $\chi = [\phi(x_1) \dots \phi(x_i) \dots \phi(x_N)]^T \in \mathbb{R}^{N \times h}$ define the data matrix in the feature space H , then C_ϕ can be expressed as:

$$C_\phi = \frac{1}{N} \chi^T \chi \quad (3)$$

The principal components of the mapped data $\phi(x_1) \dots \phi(x_i) \dots \phi(x_N)$ are computed by solving the eigenvalue decomposition of C_ϕ , such that:

$$\lambda_j \mu_j = C_\phi \mu_j \text{ with } j = 1, \dots, h \quad (4)$$

With μ_j being the j th eigenvector and λ_j the associated j th eigenvalue. For $\lambda_j \neq 0$, there exist coefficients $\alpha_{i,j}; i = 1 \dots N$, such that all eigenvectors μ_j can be considered as a linear combination of $[\phi(x_1) \phi(x_2) \dots \phi(x_N)]$ and can be expressed by:

$$\mu_j = \sum_{i=1}^N \alpha_{i,j} \phi(x_i) \quad (5)$$

However, in practice, the mapping function ϕ is not defined and then the covariance matrix C_ϕ in the feature space cannot be calculated implicitly. Thus, instead of solving the eigenvalue problem directly on C_ϕ , we apply the kernel trick firstly used for Support

Vector Machine (SVM) [25]. The inner product given in Equation (2) may be calculated by a kernel function k that satisfies Mercer's theorem [12] as follows:

$$\langle \phi(x), \phi(x') \rangle_H = k(x, x') \forall x, x' \in \mathbb{R}^m \quad (6)$$

Let us define a kernel matrix K associated with a kernel function k as:

$$K = \chi\chi^T = \begin{bmatrix} k(x_1, x_1) & \cdots & k(x_1, x_N) \\ \vdots & \ddots & \vdots \\ k(x_N, x_1) & \cdots & k(x_N, x_N) \end{bmatrix} \in \mathbb{R}^{N \times N} \quad (7)$$

Applying the kernel matrix may reduce the problem of the eigenvalue decomposition of C_ϕ [26]. Hence, eigendecomposition of the kernel matrix K is equivalent to performing PCA in \mathbb{R}^H , so that:

$$sN\Lambda V = KV \quad (8a)$$

where Λ is the diagonal matrix of eigenvalues λ_j arranged in descending order

$$\Lambda = \text{diag}(\lambda_1 \dots \lambda_j \dots \lambda_N) \quad (8b)$$

and V is the matrix of their corresponding eigenvectors.

$$V = [\alpha_1 \dots \alpha_j \dots \alpha_N] \quad (8c)$$

Since the principal components are orthonormal, it is required to guarantee the normality of μ_j in Equation (4), such that:

$$\langle \mu_j, \mu_j \rangle_H = 1; j = 1 \dots n \quad (9)$$

n is the number of the first non-zero eigenvalues.

2.1. Number of Principal Components

Determining the number of retained principal components (\downarrow) is an important step of modeling based on KPCA. The Cumulative Percent Variance (CPV) has been proposed to compute the retained PC (\downarrow) [27,28]. The cumulative percent variance (CPV) is the sum of the first \downarrow eigenvalues divided by their total variations. It can be expressed as:

$$\text{CPV}(\downarrow) = \frac{\sum_{j=1}^{\downarrow} \lambda_j}{\sum_{j=1}^m \lambda_j} 100\% \quad (10)$$

The number \downarrow of retained PCs is chosen if the CPV is higher than 95%.

2.2. Fault Detection

Like in the PCA approach, the squared prediction error (SPE) is usually used for fault detection using KPCA [29,30]. However, the conventional KPCA does not provide any approach to data reconstruction in the feature space. Thus, the computation SPE index is difficult in the KPCA method. Kim [31] and Lahdhiri [32] proposed a simple expression to calculate SPE in the feature space H , which is shown as follows:

$$\text{SPE} = \left\| k(x, x) - k_{x_t}^T \hat{P} \hat{\Lambda}^{-1} \hat{P}^T k_{x_t} \right\| \quad (11)$$

where $\hat{P} = [\alpha_1, \dots, \alpha_{\downarrow}]$ is the matrix of the first \downarrow principal eigenvectors of K , $\hat{\Lambda} = \text{diag}[\lambda_1, \dots, \lambda_{\downarrow}]$ is the diagonal matrix of the first \downarrow eigenvalues of K [33], and $k_{x_t} = [k(x_1, x_t), \dots, k(x_N, x_t), \dots, k(x_N, x_t)]^T; i = 1 \dots N$.

The confidence limit for the SPE index can be calculated using the χ^2 distribution and is given by:

$$\text{SPE}_{\text{lim}} \approx \delta_{\alpha}^2 \quad (12)$$

where δ_{α}^2 is the control limit expressed by:

$$\delta_{\alpha}^2 = g\chi_{h,\alpha}^2 \quad (13)$$

with: $g = \frac{b}{2a}$ and $h = \frac{2a^2}{b}$, where a is the estimated mean and b is the variance of the SPE [34,35].

3. The Observer Method

In fault diagnosis, an observer is a mathematical model or algorithm used to estimate the state variables and fault parameters of a system based on available measurements. The structure of an observer varies depending on the type of system being observed and the nature of the faults being diagnosed. However, the general structure of an observer typically involves the following components:

- **System Model:** The observer relies on a mathematical model that describes the dynamics of the system being observed. This model can be derived from first principles or obtained through system identification techniques.
- **Measurement Equation:** The observer uses a measurement equation that relates the system's state variables to the available measurements. This equation can be derived from the system model and typically includes sensor equations and/or sensor noise models.
- **State Estimation:** The core of the observer is the state estimation algorithm, which updates and estimates the system's state variables based on the available measurements. Various estimation techniques can be used, such as Kalman filters, extended Kalman filters, particle filters, or model-based observers like the sliding mode observer.
- **Fault Detection:** In fault diagnosis, the observer is also responsible for detecting the occurrence of faults. This can be done by comparing the estimated state variables with expected values or by analyzing the residuals between the measurements and the estimated values.
- **Fault Parameter Estimation:** If faults are detected, the observer may also estimate the fault parameters, such as fault magnitudes, locations, or characteristics. This is typically done by incorporating fault models into the observer and updating the estimated parameters based on the available measurements and fault detection results.
- **Adaptation and Learning:** Depending on the observer's design, it may incorporate adaptation or learning mechanisms to improve its performance over time. These mechanisms allow the observer to adapt to changes in system dynamics or fault characteristics, or to learn from historical data to enhance its fault diagnosis capabilities.

It is important to note that the specific structure and algorithm used for fault diagnosis observers can vary greatly depending on the application, system complexity, and available information. Different domains, such as power systems, automotive, or aerospace, may have specialized observer designs tailored to their specific requirements. A system with p inputs denoted $u(t)$ and m output measurements denoted $x(t)$.

The dynamic behavior of this system is described by the following equations [36]:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (14)$$

$$\begin{aligned} y(t) &= Cx(t) \\ x(0) &= x_0 \end{aligned} \quad (15)$$

where $x(t) \in R^n$ is the state vector, Ref. [37] is the output vector, and $u(t) \in R^r$ is the input vector.

Note that matrices A , B , and C represent the state-space description of a linear time-invariant system [38,39], and they have appropriate dimensions with those of the vectors $x(t)$, $u(t)$, and $y(t)$.

Given that the state is not generally available [40], the objective is an observer in order to perform a feedback control condition and estimate this state by a variable which we denote as $\hat{x}(t)$. This estimate is carried out by a dynamic system, the output will be precisely $\hat{x}(t)$, and the input will consist of all the information available [41], that is to say, $u(t)$ and $y(t)$. The structure of an observer can be written as:

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + Bu(t) + K(y(t) - \hat{y}(t)) \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \tag{16}$$

where the correction term appears clearly in terms of the reconstruction error of the output $y(t) - \hat{y}(t)$, and the correct term can be written as an ion gain [42], L , or the determined observer gain [43]. This structure can be written as:

$$\dot{\hat{x}}(t) = (A - KC)\hat{x}(t) + Bu(t) + Ky(t) \tag{17}$$

If we consider the estimation error: $\tilde{x}(t) = x(t) - \hat{x}(t)$, we obtain: $\dot{\tilde{x}}(t) = (A - KC)\tilde{x}(t)$.

The observer described by Equation (15) is illustrated in Figure 1.

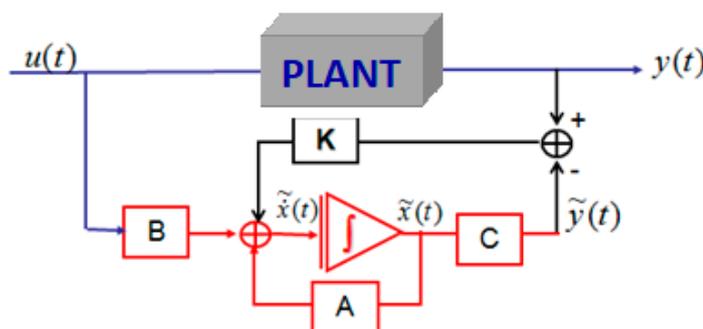


Figure 1. Structure of the observer.

Fault Detection Observer

In this work, we assume that the actuators and sensors are affected by faults. Our goal is to detect and isolate the faults [43]. The state of the system model can be written as:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B(u(t) + f_a(t)) \\ y(t) &= Cx(t) + f_c(t) \end{aligned} \tag{18}$$

where $f_a(t)$ is the actuator fault of the actuators and $f_c(t)$ is a sensor fault.

$$\begin{aligned} \dot{\hat{x}}(t) &= (A - KC)\hat{x}(t) + Bu(t) + Ky(t) \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \tag{19}$$

where $\hat{x}(t) \in R^n$ is an estimated state vector, K is a matrix such that $A - KC$ is stable and its proper values have a real part smaller than A . This leads to:

$$\begin{aligned} \dot{\tilde{x}}(t) &= (A - KC)\tilde{x}(t) + Bf_a(t) - Kf_c(t) \\ \tilde{y}(t) &= C\tilde{x}(t) + f_c(t) \end{aligned} \tag{20}$$

If K is such that $A - KC$ is a Hurwitz matrix, the residue $\tilde{y}(t)$ tends to 0 well in the absence of defects. Transfer between faults and residuals can be written:

$$\tilde{y}(t) = C(pI - A + KC)^{-1}Bf_a(t) + \left[I - C(pI - A + KC)^{-1}K \right] f_c(t) \quad (21)$$

where p is a temporal derivative operator.

Which leads to [44], taking into account the inversion lemma in:

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DC^{-1}B)^{-1}DA^{-1} \quad (22)$$

which can be written as:

$$\left[I + C(pI - A)^{-1}K \right] \tilde{y}(t) = C(pI - A)^{-1}Bf_a(t) + f_c(t) \quad (23)$$

From this relationship and in the absence of actuator failure, the system for isolating fault sensors from residues can be written as:

$$\dot{\eta}(t) = A\eta(t) + K\tilde{y}(t) \quad (24)$$

$$f_a(t) = C\eta(t) + \tilde{y}(t) \quad (25)$$

This variable is estimated using the same system as before, that is to say:

$$\begin{aligned} \dot{\eta}(t) &= A\eta(t) + K\tilde{y}(t) \\ \zeta(t) &= C\eta(t) + \tilde{y}(t) \end{aligned} \quad (26)$$

And the estimation of sensor failure is given by the inversion of the initial model of the system:

$$\beta(t) = \Delta(p) \left[C(pI - A)^{-1}B \right]^{-1} \zeta(t) \quad (27)$$

where $\Delta(p) = \text{diag} \left\{ \frac{1}{(p+\alpha)^i} \right\}$ is the matrix filters as $\Delta(p) \left[C(pI - A)^{-1}B \right]^{-1}$ is bi-causal [45].

$\beta(t)$ is not an estimate of faults but rather filtering defects; however, the character of diagonal $\Delta(p)$ allows the isolation [20].

4. Applications

4.1. Overview of Three-Tank System Applications

A three-tank water system, also referred to as a triple-tank or multi-tank system, exhibits a water storage and distribution network, using multiple tanks for various functions. Specific uses for three-tank water systems vary, but typically fall within environmental engineering, including Rainwater Harvesting and Reuse systems, greywater recycling, and off-grid water supply [46–48]. These applications have found their place in the heart of targeting sustainable practices, more specifically SDG6, by ensuring Clean Water and Sanitation [46]. A three-tank system can be employed for rainwater harvesting and reuse, with each tank serving its own specific function. One tank could collect rainwater from the roof, another could be dedicated to filtering and treating (primary, secondary, and tertiary water treatment processes), while a third could store treated rainwater for urban reuse in fields such as irrigation. This system helps decrease the reliance on freshwater sources while supporting sustainable water management practices [46].

Triple tank systems provide an effective method of greywater recycling via treating and reusing urban residual water, generated from domestic activities like bathing, handwashing, and laundry. One tank collects urban water for disinfection processes (mostly chlorine and combined chlorine sub-products) [49], while a second tank stores treated water for reuse, in activities such as toilet flushing or landscape irrigation. This approach helps release the stress on freshwater resources while alleviating strain on sewerage systems [47].

In remote or off-grid locations, the availability of reliable water resources is limited. Hence, a three-tank water system can serve as a reliable self-sufficient water supply source. One tank could hold water collected from natural sources like wells or springs, while another would treat and purify it before the third would store the treated water for domestic consumption [48].

On the other hand, several shortcomings and limitations may arise from the installation of three-tank systems. Following the sub-division of the system into three counterparts, the urge for complexity and space requirements are more demanding [46–48]. Additional space must also be implemented, as well as the multiple features required to connect the tanks (pumps, valves, etc.). In addition, this system exhibits low cost-effectiveness, as it involves high installation prices compared to the simple water storage system [50]. The maintenance and additional unit operations, required for water screening, treatment, and distribution, will incur additional expenses [51]. Following the complexity of the system, more design heuristics and trade-offs should be considered. This will make the maintenance operation more tedious and time-consuming [51]. On the human factor and engineering intuition level, a high level of skill and knowledge of handling are required, for the sake of preserving the integrity of the system [46–48].

Most of the aforementioned factors could be overcome by implementing a proper control system, based on the fact that multiple error sources could be generated. KPCA exhibits a highly convenient unsupervised machine learning approach for a three-tank system control, as it involves dealing with intercorrelated data input [52]. In other words, the error in one part of the system will definitely influence the other components.

4.2. Process Description of a Hydraulic System with Three Tanks

As illustrated in Figure 2, the considered process is a three-tank system with two inputs and three outputs. It consists of three tanks with identical sections, supplied with distilled water. They are serially interconnected by two cylindrical pipes with identical sections [53,54]. The pipes of communication between tanks T_1 and T_2 are equipped with manually adjustable valves; the flow rates of the connection pipes can be controlled using ball valves a_{z1} and a_{z2} . The plant has one outlet pipe located at the bottom of tank T_3 . There are three other pipes each installed at the bottom of each tank, which are provided with a direct connection (outflow rate) to the reservoir with ball valves b_{z1} , b_{z2} , and b_{z3} , respectively. The pipes can only be manipulated manually [12]. Pumps 1 and 2 are supplied by water from the reservoir with flow rates $Q_1(t)$ and $Q_2(t)$, respectively. The necessary level measurements $h_1(t)$, $h_2(t)$, and $h_3(t)$ are carried out by the piezo-resistive differential pressure sensors.

The state equations are obtained by writing that the variation of the water volume in a tank is equal to the difference between the incoming flow and the outgoing flows which means the water of tanks 1 and 2 can flow toward tank 3.

Then, the system can be represented by the following equations:

$$\dot{h}_i(t) = \frac{1}{A}(Q_i^{in}(t) - Q_{ij}^{out1}(t) - Q_{ij}^{out2}(t)), i, j = 1, 2, 3 \quad (28)$$

where $Q_i^{in}(t)$ is the flow through pump i ($i = 1; 2$), and $Q_{ij}^{out1}(t)$ represents the flow rates of water between tanks i and j ($i, j = 1, 2, 3 \forall i \neq j$), and can be expressed using the law of Torricelli [29].

$$Q_{ij}^{out1}(t) = a_{zi} S_n \text{sign}(h_i - h_j) \sqrt{2g|h_i - h_j|}, j = 1, 3 \quad (29)$$

and $Q_{ij}^{out2}(t)$ represents the outflow rate, given by:

$$Q_{ij}^{out2}(t) = b_{zj} S_L \sqrt{2gh_i} \quad j = 1, 2, 3 \quad (30)$$

where $h_i(t)$, $Q_i^{in}(t)$, and $Q_{ij}^{out}(t)$ are, respectively, the levels of water, the input flow rate, and the output flow rate.

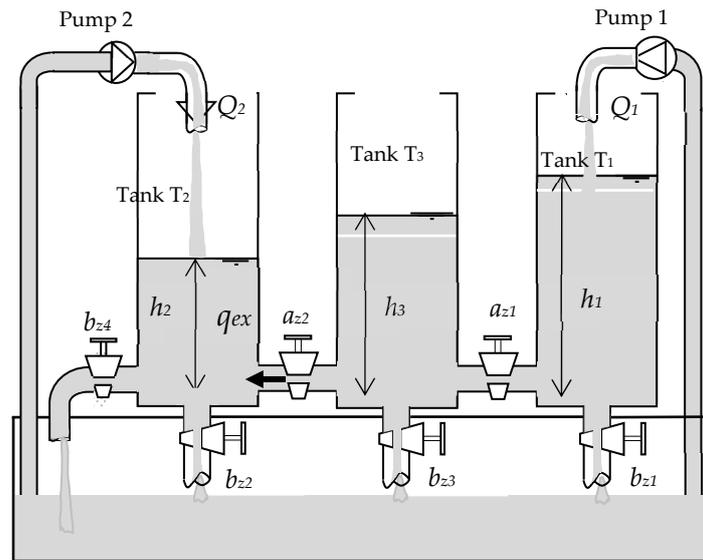


Figure 2. Three-tank system.

The parameters of the three-tank system are defined as follows. The controlled signals are the water levels (h_2, h_3) of tank 2 and tank 3. These levels are controlled by two pumps. The system can be considered as a multi-input multi-output system (MIMO) [54], where the input is inflow rates Q_1 and Q_2 and the output is liquid levels h_2 and h_3 . Then the three-tank system can be modeled by the following three differential equations:

$$\begin{cases} \frac{dh_1}{dt} = -c_1 \text{sign}(h_1 - h_3) \sqrt{|h_1 - h_3|} - B_1 \sqrt{h_1} + \frac{Q_1}{a} \\ \frac{dh_2}{dt} = c_3 \text{sign}(h_3 - h_2) \sqrt{|h_3 - h_2|} - (B_4 + B_2) \sqrt{h_2} + \frac{Q_2}{a} \\ \frac{dh_3}{dt} = c_1 \text{sign}(h_1 - h_3) \sqrt{|h_1 - h_3|} - (B_3 + B_2) \sqrt{h_3} - c_3 \text{sign}(h_3 - h_2) \sqrt{|h_3 - h_2|} \end{cases} \quad (31)$$

where the parameters $c_i, i = 1, 3$ and $B_j, j = 1, 2, 3, 4$ are defined by:

$$c_i = \frac{1}{A} a_{zi} S_n \sqrt{2g} i = 1, 3 \quad (32)$$

$$B_j = \frac{1}{A} b_{zj} S_L \sqrt{2g} j = 1, 2, 3, 4 \quad (33)$$

While taking $B_1 = B_2 = B_3 = 0$, the three equations of the system become:

$$\begin{cases} \frac{dh_1}{dt} = -c_1 \text{sign}(h_1 - h_3) \sqrt{|h_1 - h_3|} + \frac{Q_1}{a} \\ \frac{dh_2}{dt} = c_3 \text{sign}(h_3 - h_2) \sqrt{|h_3 - h_2|} - B_4 \sqrt{h_2} + \frac{Q_2}{a} \\ \frac{dh_3}{dt} = c_1 \text{sign}(h_1 - h_3) \sqrt{|h_1 - h_3|} - c_3 \text{sign}(h_3 - h_2) \sqrt{|h_3 - h_2|} \end{cases} \quad (34)$$

At equilibrium, for a constant water level set point, the level derivatives must be zero.

$$\dot{h}_1 = \dot{h}_2 = \dot{h}_3 = 0 \quad (35)$$

Therefore, using (31) in the steady state, the following algebraic relationship holds.

$$\begin{cases} -c_1 \text{sign}(h_1 - h_3) \sqrt{|h_1 - h_3|} + \frac{Q_1}{a} = 0 \\ c_1 \text{sign}(h_1 - h_3) \sqrt{|h_1 - h_3|} - c_3 \text{sign}(h_3 - h_2) \sqrt{|h_3 - h_2|} = 0 \end{cases} \quad (36)$$

For the coupled-tank system, the fluid flow Q_1 into tank 1 cannot be negative because the pump can only drive water into the tank, then:

$$Q_1 \geq 0$$

From (36), we have:

$$c_1 \text{sign}(h_1 - h_3) \sqrt{|h_1 - h_3|} = \frac{Q_1}{a} \quad (37)$$

and $c_1 \text{sign}(h_1 - h_3) \sqrt{|h_1 - h_3|} = c_3 \text{sign}(h_3 - h_2) \sqrt{|h_3 - h_2|}$.

Then, $(h_1 - h_3) \geq 0$ and $(h_3 - h_2) \geq 0$. Therefore, if we assume

$$x_1 = h_1, x_2 = h_2, x_3 = h_3, u_1 = Q_1 \text{ and } u_2 = Q_2 \quad (38)$$

We have

$$\begin{cases} \dot{x}_1 = -c_1 \sqrt{x_1 - x_3} + \frac{u_1}{a} \\ \dot{x}_2 = c_3 \sqrt{x_3 - x_2} - B_4 \sqrt{x_2} + \frac{u_2}{a} \\ \dot{x}_3 = c_1 \sqrt{x_1 - x_3} - c_3 \sqrt{x_3 - x_2} \end{cases} \quad (39)$$

Which can be written as:

$$\begin{cases} \dot{x} = f(t, x) + gu \\ y = cx \end{cases} \quad (40)$$

where

$$x = [x_1 \quad x_2 \quad x_3]^T, u = [u_1 \quad u_2]^T, y = [x_2 \quad x_3]^T \quad (41)$$

$$f(t, x) = \begin{pmatrix} -c_1 \sqrt{x_1 - x_3} \\ c_3 \sqrt{x_3 - x_2} - B_4 \sqrt{x_2} \\ c_1 \sqrt{x_1 - x_3} - c_3 \sqrt{x_3 - x_2} \end{pmatrix}, g = \begin{pmatrix} \frac{1}{a} & 0 \\ 0 & \frac{1}{a} \\ 0 & 0 \end{pmatrix} \text{ and } c = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (42)$$

4.3. Simulation Results

We are interested in the fault detection of the pressure sensors, which measure the water levels (h_2, h_3) of tank 2 and tank 3 using conventional KPCA and the observer method. A total of 5000 samples were generated from this process [55]. The 1000 first samples were used to construct the KPCA model; the last 3000 samples are used to test the fault detection methods. We have used the Radial Basis Function (RBF) [56,57]. Two types of faults are considered [58]: faults in the pressure sensor of tank 2 and faults in the pressure sensor of tank 3.

Case 1: faults in the pressure sensor of tank 2 (water level h_2).

Fault 1: a step bias of h_2 by adding 10% more than its range of variation [59]. The fault is introduced between samples 1500 and 3000.

The SPE index is a statistical measure commonly used in multivariate analysis or process monitoring. It quantifies the discrepancy between predicted and observed values in a model. In this context, Figure 3 represents the results of an analysis or experiment involving a fault (Fault 1) and the SPE (Squared Prediction Error) index, with the variable h_2 being affected. The evolution indicates that the figure shows changes or trends over time or some other continuous parameter. It suggests that the data or analysis captured a dynamic process or progression.

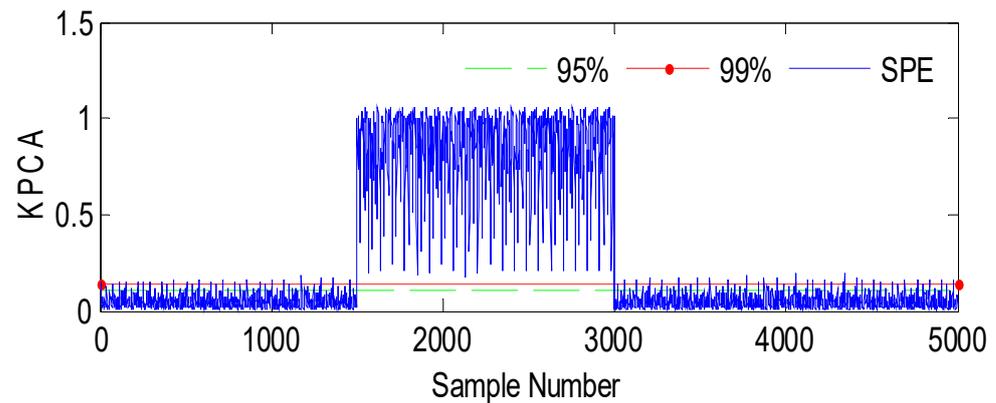


Figure 3. Evolution of the SPE index with Fault 1 affecting h_2 using KPCA.

Fault 2: a step bias of h_2 by adding 20% more than its range of variation. The fault is introduced between samples 3500 and 4500.

Case 2: Faults in the pressure sensor of tank 3 (water level h_3)

Fault 1: a step bias of h_3 by adding 10% more than its range of variation. The fault is introduced between samples 2000 and 3000 [60].

Fault 2: a step bias of h_3 by adding 20% more than its range of variation. The fault is introduced between samples 3000 and 4000.

Based on the observations from Figures 4–10, it is evident that both the kernel method and the observer technique detect a fault occurrence, as indicated by the defective outputs h_2 and h_3 . Both methods yield effective and comparable results in terms of sensor fault detection. In conclusion, the simulation results using the “observer-based model and a kernel technique called KPCA” show that these two techniques are comparable in the field of fault diagnosis. These methods have been evaluated and demonstrated similar performances in terms of fault identification and detection. This suggests that the use of an observer-based model and the KPCA technique can be effective in diagnosing faults in a system. However, it is important to note that the comparison of performance between these techniques may depend on the specific context of the application and the characteristics of the system being studied. Further studies and experimental tests may be necessary to confirm these results and assess their applicability in other domains.

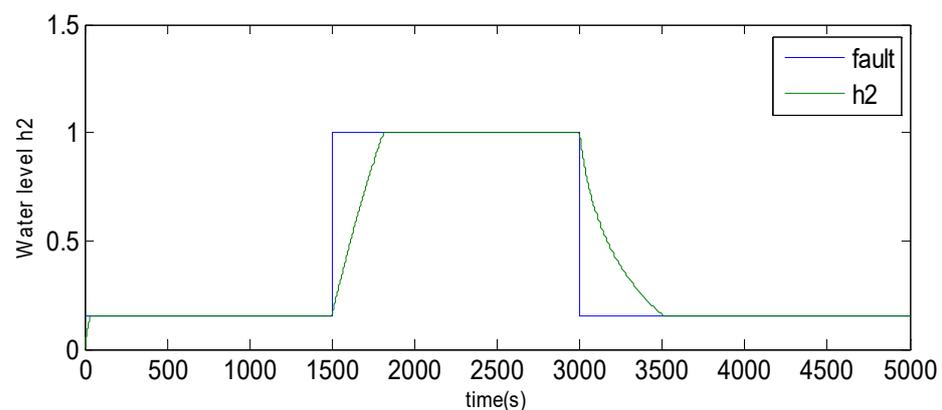


Figure 4. Evolution of sensor h_2 between 1500 and 3000 using the observer method.

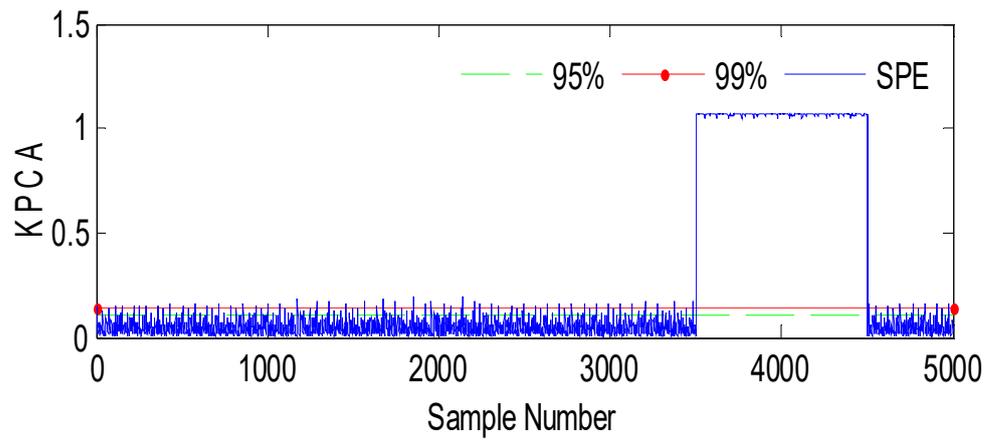


Figure 5. Evolution of the SPE index with Fault 2 affecting h_2 using KPCA.

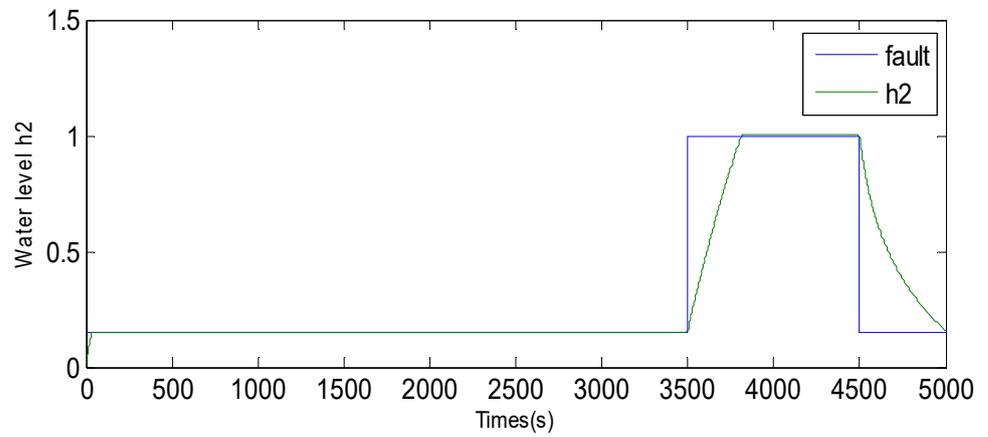


Figure 6. Evolution of sensor h_2 between 3500 and 4500 using the observer method.

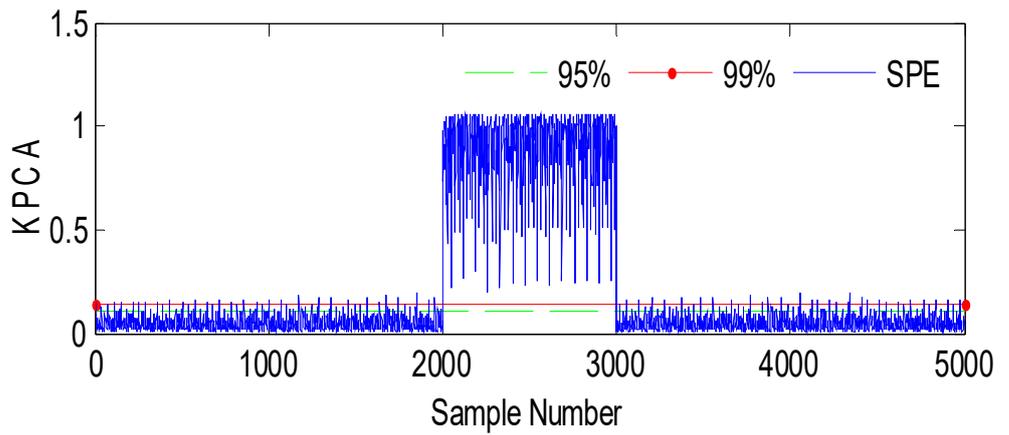


Figure 7. Evolution of the SPE index with Fault1 affecting h_3 using KPCA.

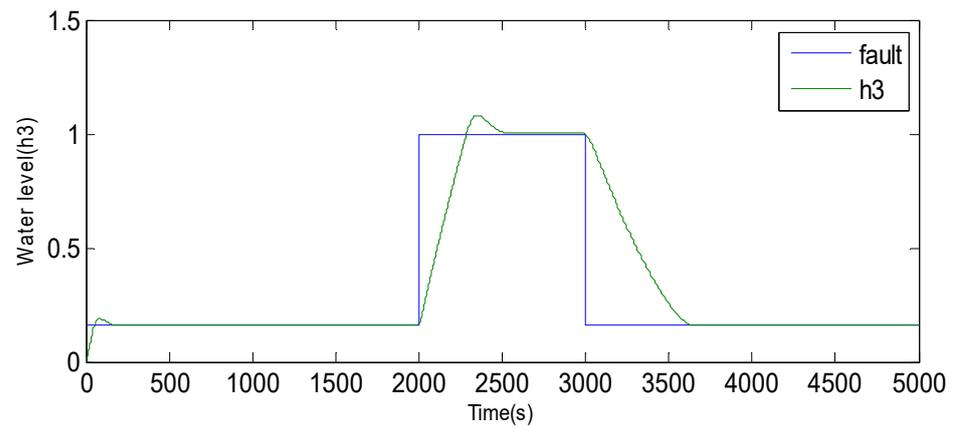


Figure 8. Evolution of sensor h_3 between 2000 and 3000 using the observer method.

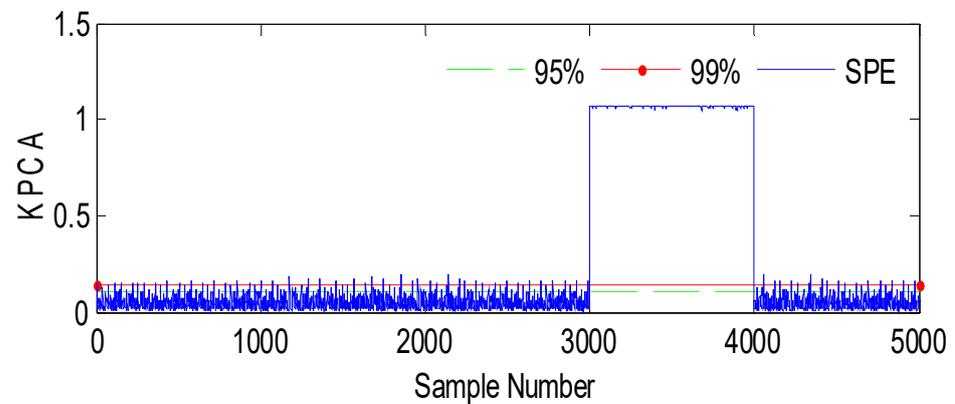


Figure 9. Evolution of the SPE index with Fault 2 affecting h_3 using KPCA.

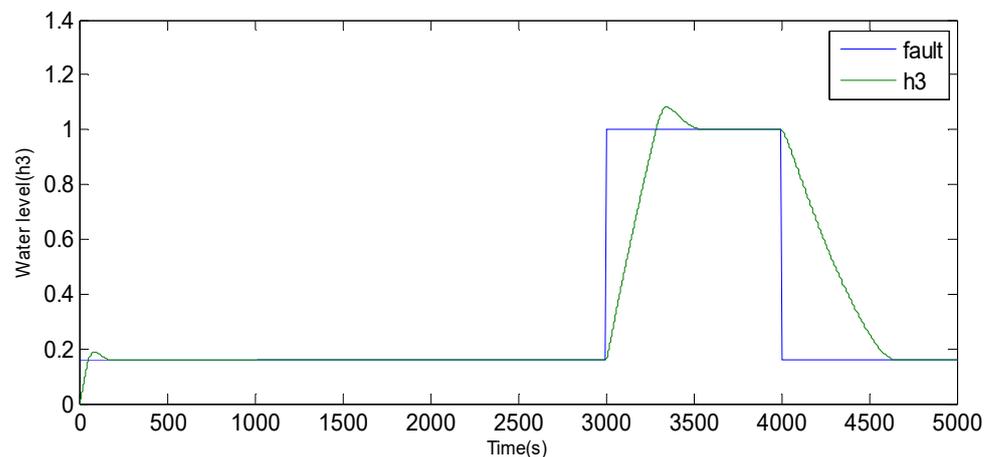


Figure 10. Evolution of sensor h_3 between 3000 and 4000 using the observer method.

5. Conclusions

This paper offers a comprehensive examination of an observer-based model and a kernel technique called KPCA, both employed for sensor fault detection [61]. The observer's operational principle is thoroughly explained, providing a detailed understanding of its function. To assess the effectiveness of these two techniques, a comparative analysis is conducted using a three-tank process. The simulation results yield valuable insights, demonstrating that both the observer-based model and the KPCA method yield satisfactory results. However, it is important to note that these findings are based on a single case study and cannot be considered in isolation. To establish the broader applicability of these

techniques across various systems and conditions, further investigation is necessary. The technique used to validate our system is the injection of faults and we observe the reaction of the system with modification of the time each time. The way of identifying faults is validated and can be used in industry, specifically in chemical systems.

The problem is the difficulty of detecting defects. Our method has approved the capability of detecting defects despite the change in the nature of defects and the integration time.

Consequently, additional research is planned to advance the understanding of fault detection techniques through a deeper exploration of these approaches.

“Kernel Principal Component Analysis (KPCA) and observer-based approaches are two distinct methods commonly used in fault detection within the realms of machine learning and control theory. KPCA is a nonlinear dimensionality reduction technique that maps data into a high-dimensional feature space using kernel functions, enabling the detection of faults in complex and nonlinear systems. Its advantages lie in its ability to handle nonlinearity, making it suitable for intricate processes. However, KPCA’s performance heavily relies on the appropriate choice of kernel and its associated parameters, which can be challenging to determine in practice. On the other hand, observer-based approaches leverage mathematical models to estimate the system’s behavior and compare it with the actual response for detecting faults. The advantage of observer-based methods is their inherent robustness to disturbances and noise, allowing them to perform well in noisy environments. Nevertheless, these approaches often require a comprehensive and accurate model of the system, and their performance may suffer if the model is not precise or if the system exhibits significant nonlinear behavior. Overall, choosing between KPCA and an observer-based approach depends on the specific characteristics of the system, the availability of accurate models, and the level of nonlinearity present, as each method offers distinct strengths and weaknesses in fault detection applications”.

Author Contributions: Methodology, F.L. and L.M.; Validation, W.A.; Formal analysis, H.D.; Writing—review & editing, K.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Lima, A.; Zen, H.; Nankaku, Y.; Tokuda, K.; Kitamura, T.; Resende, F.G. Applying Sparse KPCA for Feature Extraction in Speech Recognition. *IEICE Trans. Inf. Syst.* **2005**, *88*, 401–409. [[CrossRef](#)]
2. Aronszajn, N. Theory of Reproducing Kernels. *Trans. Am. Math. Soc.* **1950**, *68*, 337–404. [[CrossRef](#)]
3. Baklouti, R.; Mansouri, M.; Nounou, M.; Nounou, H.; Hamida, A.B. Iterated Robust Kernel Fuzzy Principal Component Analysis and Application to Fault Detection. *J. Comput. Sci.* **2016**, *15*, 34–49. [[CrossRef](#)]
4. Ratsch, G. Input Space versus Feature Space in Kernel-Based Methods. *IEEE Trans. Neural Netw.* **1999**, *10*, 1000–1017.
5. 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](#)]
6. Chetouani, Y. A Neural Network Approach for the Real-Time Detection of Faults. *Stoch. Environ. Res. Risk Assess.* **2008**, *22*, 339–349. [[CrossRef](#)]
7. Cho, J.-H.; Lee, J.-M.; Choi, S.W.; Lee, D.; Lee, I.-B. Fault Identification for Process Monitoring Using Kernel Principal Component Analysis. *Chem. Eng. Sci.* **2005**, *60*, 279–288. [[CrossRef](#)]
8. 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](#)]
9. Doan, P.T.; Bui, T.L.; Kim, H.K.; Kim, S.B. Sliding-Mode Observer Design for Sensorless Vector Control of AC Induction Motor. In Proceedings of the 2013 9th Asian Control Conference (ASCC), Istanbul, Turkey, 23–26 June 2013; pp. 1–5.

10. Ruderman, M.; Iwasaki, M. Sensorless Control of Motor Velocity in Two-Mass Actuator Systems with Load Sensing Using Extended State Observer. In Proceedings of the 2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Besacon, France, 8–11 July 2014; pp. 360–365.
11. Heredia, G.; Ollero, A. Sensor Fault Detection in Small Autonomous Helicopters Using Observer/Kalman Filter Identification. In Proceedings of the 2009 IEEE International Conference on Mechatronics, Malaga, Spain, 14–17 April 2009; pp. 1–6.
12. Isermann, R. *Fault-Diagnosis Systems. An Introduction from Fault Detection to Fault Tolerance*; Springer: Berlin/Heidelberg, Germany, 2006. [\[CrossRef\]](#)
13. Li, X.-J.; Yang, G.-H. Dynamic Observer-Based Robust Control and Fault Detection for Linear Systems. *IET Control Theory Appl.* **2012**, *6*, 2657–2666. [\[CrossRef\]](#)
14. Kumar, V.E.; Jerome, J. Sensor Fault Detection in DC Servo System Using Unknown Input Observer with Structured Residual Generation. *J. Electr. Eng.* **2013**, *13*, 6.
15. Yi, J.; Huang, Z.; Liu, W.; Yang, Y.; Zhang, X.; Liu, J. Actuator Fault Detection Based on Robust Adaptive Observer for CCBII Braking System. In Proceedings of the 26th Chinese Control and Decision Conference (2014 CCDC), Changsha, China, 31 May–2 June 2014; pp. 2841–2846.
16. Khalid, H.M.; Khoukhi, A.; Al-Sunni, F.M. Fault Detection and Classification Using Kalman Filter and Genetic Neuro-Fuzzy Systems. In Proceedings of the 2011 Annual Meeting of the North American Fuzzy Information Processing Society, El Paso, TX, USA, 18–20 March 2011; pp. 1–6.
17. Yang, S.K.; Liu, T.S. State Estimation for Predictive Maintenance Using Kalman Filter. *Reliab. Eng. Syst. Saf.* **1999**, *66*, 29–39. [\[CrossRef\]](#)
18. Alkaya, A.; Eker, İ. Luenberger Observer-Based Sensor Fault Detection: Online Application to DC Motor. *Turk. J. Electr. Eng. Comput. Sci.* **2014**, *22*, 363–370. [\[CrossRef\]](#)
19. Tarantino, R.; Szigeti, F.; Colina-Morles, E. Generalized Luenberger Observer-Based Fault-Detection Filter Design: An Industrial Application. *Control Eng. Pract.* **2000**, *8*, 665–671. [\[CrossRef\]](#)
20. Schölkopf, B.; Smola, A.; Müller, K.-R. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Comput.* **1998**, *10*, 1299–1319. [\[CrossRef\]](#)
21. Dong, D.; McAvoy, T.J. Nonlinear Principal Component Analysis—Based on Principal Curves and Neural Networks. *Comput. Chem. Eng.* **1996**, *20*, 65–78. [\[CrossRef\]](#)
22. 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. [\[CrossRef\]](#)
23. Harkat, M.-F.; Mourot, G.; Ragot, J. An Improved PCA Scheme for Sensor FDI: Application to an Air Quality Monitoring Network. *J. Process Control* **2006**, *16*, 625–634. [\[CrossRef\]](#)
24. Ge, Z.; Yang, C.; Song, Z. Improved Kernel PCA-Based Monitoring Approach for Nonlinear Processes. *Chem. Eng. Sci.* **2009**, *64*, 2245–2255. [\[CrossRef\]](#)
25. Vapnik, V. The Support Vector Method of Function Estimation. In *Nonlinear Modeling: Advanced Black-Box Techniques*; Springer: Berlin/Heidelberg, Germany, 1998; pp. 55–85.
26. Harkat, M.-F.; Mourot, G.; Ragot, J. Multiple Sensor Fault Detection and Isolation of an Air Quality Monitoring Network Using RBF-NLPCA Model. *IFAC Proc. Vol.* **2009**, *42*, 828–833. [\[CrossRef\]](#)
27. 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. [\[CrossRef\]](#)
28. 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. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Kallas, M.; Mourot, G.; Maquin, D.; Ragot, J. Diagnosis of Nonlinear Systems Using Kernel Principal Component Analysis. *Proc. J. Phys. Conf. Ser.* **2014**, *570*, 072004. [\[CrossRef\]](#)
30. Kazor, K.; Holloway, R.W.; Cath, T.Y.; Hering, A.S. Comparison of Linear and Nonlinear Dimension Reduction Techniques for Automated Process Monitoring of a Decentralized Wastewater Treatment Facility. *Stoch. Environ. Res. Risk Assess.* **2016**, *30*, 1527–1544. [\[CrossRef\]](#)
31. Kim, K.I.; Jung, K.; Kim, H.J. Face Recognition Using Kernel Principal Component Analysis. *IEEE Signal Process. Lett.* **2002**, *9*, 40–42.
32. Lahdhiri, H.; Taouali, O.; Elaissi, I.; Jaffel, I.; Harakat, 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. [\[CrossRef\]](#)
33. Lefebvre, D.; Basile, F. An Approach Based on Timed Petri Nets and Tree Encoding to Implement Search Algorithms for a Class of Scheduling Problems. *Inf. Sci.* **2021**, *559*, 314–335. [\[CrossRef\]](#)
34. Lajmi, F.; Talmoudi, A.J.; Dhouibi, H. Fault Diagnosis of Uncertain Systems Based on Interval Fuzzy PETRI Net. *Stud. Inform. Control* **2017**, *26*, 239–248. [\[CrossRef\]](#)
35. Fatma, L.; Ghabi, J.; Dhouibi, H. Applying Interval Fuzzy Petri Net to Failure Analysis. *Int. J. Serv. Sci. Manag. Eng. Technol.* **2020**, *11*, 14–30.
36. Lee, J.-M.; Yoo, C.; Lee, I.-B. Statistical Process Monitoring with Independent Component Analysis. *J. Process Control* **2004**, *14*, 467–485. [\[CrossRef\]](#)

37. 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](#)]
38. Li, G.; Qin, S.J.; Zhou, D. Geometric Properties of Partial Least Squares for Process Monitoring. *Automatica* **2010**, *46*, 204–210. [[CrossRef](#)]
39. Li, H.; Zhang, D. Stochastic Representation and Dimension Reduction for Non-Gaussian Random Fields: Review and Reflection. *Stoch. Environ. Res. Risk Assess.* **2013**, *27*, 1621–1635. [[CrossRef](#)]
40. 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](#)]
41. Aizerman, A. Theoretical Foundations of the Potential Function Method in Pattern Recognition Learning. *Autom. Remote Control* **1964**, *25*, 821–837.
42. 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](#)]
43. Mercer, J. XVI. Functions of Positive and Negative Type, and Their Connection the Theory of Integral Equations. *Philos. Trans. R. Soc. Lond. Ser. A Contain. Pap. A Math. Phys. Character* **1909**, *209*, 415–446.
44. Mika, S.; Schölkopf, B.; Smola, A.; Müller, K.-R.; Scholz, M.; Rätsch, G. Kernel PCA and De-Noising in Feature Spaces. In *Proceedings of the Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 1998; Volume 11.
45. Nomikos, P.; MacGregor, J.F. Multivariate SPC Charts for Monitoring Batch Processes. *Technometrics* **1995**, *37*, 41–59. [[CrossRef](#)]
46. Teston, A.; Piccinini Sclaro, T.; Kuntz Maykot, J.; Ghisi, E. Comprehensive Environmental Assessment of Rainwater Harvesting Systems: A Literature Review. *Water* **2022**, *14*, 2716. [[CrossRef](#)]
47. Al-Jayyousi, O.R. Greywater Reuse: Towards Sustainable Water Management. *Desalination* **2003**, *156*, 181–192. [[CrossRef](#)]
48. Bahta, S.T. Design and Analyzing of an Off-Grid Hybrid Renewable Energy System to Supply Electricity for Rural Areas: Case Study: Atsbi District, North Ethiopia. Master's Thesis, KTH School of Industrial Engineering and Management, Stockholm, Sweden, 2013.
49. Cheremisinoff, P.N. *Handbook of Water and Wastewater Treatment Technology*; Routledge: London, UK, 2019.
50. Bojan-Dragos, C.-A.; Szedlak-Stinean, A.-I.; Precup, R.-E.; Gurgui, L.; Hedrea, E.-L.; Mituletu, I.-C. Control Solutions for Vertical Three-Tank Systems. In *Proceedings of the 2018 IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, Timisoara, Romania, 17–19 May 2018; pp. 000593–000598.
51. Peters, M.S.; Timmerhaus, K.D.; West, R.E. *Plant Design and Economics for Chemical Engineers*; McGraw-Hill: New York, NY, USA, 2003; Volume 4.
52. Jolliffe, I. Principal Component Analysis. In *Encyclopedia of Statistics in Behavioral Science*; Wiley Online Library: New York, NY, USA, 2005.
53. Sheriff, M.Z.; Mansouri, M.; Karim, M.N.; Nounou, H.; Nounou, M. Fault Detection Using Multiscale PCA-Based Moving Window GLRT. *J. Process Control* **2017**, *54*, 47–64. [[CrossRef](#)]
54. Taouali, O.; Elaissi, I.; Messaoud, H. Dimensionality Reduction of RKHS Model Parameters. *ISA Trans.* **2015**, *57*, 205–210. [[CrossRef](#)] [[PubMed](#)]
55. Tharrault, Y.; Mourot, G.; Ragot, J.; Maquin, D. Fault Detection and Isolation with Robust Principal Component Analysis. *Int. J. Appl. Math. Comput. Sci.* **2008**, *18*, 429–442. [[CrossRef](#)]
56. Vapnik, V.N. An Overview of Statistical Learning Theory. *IEEE Trans. Neural Netw.* **1999**, *10*, 988–999. [[CrossRef](#)] [[PubMed](#)]
57. Zhang, Y.; Li, S.; Teng, Y. Dynamic Processes Monitoring Using Recursive Kernel Principal Component Analysis. *Chem. Eng. Sci.* **2012**, *72*, 78–86. [[CrossRef](#)]
58. Zhao, C.; Ma, J.; Wei, Y.; Long, Y.; Ou, H.; Bao, J.; Yin, J.; Liu, W.; Zhu, N.; Lu, X. Active Mass Transfer for Printable Electrochemical Sensors with Ultrasonic Stimuli. *Mater. Today Commun.* **2023**, *34*, 105382. [[CrossRef](#)]
59. Magni, J.-F.; Mouyon, P. A Generalized Approach to Observers for Fault Diagnosis. In *Proceedings of the 30th IEEE Conference on Decision and Control*, Brighton, UK, 11–13 December 1991; pp. 2236–2241. [[CrossRef](#)]
60. Zhong, M.; Yang, Y.; Sun, S.; Zhou, Y.; Postolache, O.; Ge, Y. Priority-Based Speed Control Strategy for Automated Guided Vehicle Path Planning in Automated Container Terminals. *Trans. Inst. Meas. Control* **2020**, *42*, 014233122094011. [[CrossRef](#)]
61. Wu, P.; Xiao, F.; Sha, C.; Huang, H.; Sun, L. Trajectory Optimization for UAVs' Efficient Charging in Wireless Rechargeable Sensor Networks. *IEEE Trans. Veh. Technol.* **2020**, *69*, 4207–4220. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.