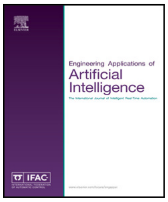
Engineering Applications of Artificial Intelligence 128 (2024) 107510

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



A derivative, integral, and proportional features extractor for fault detection in dynamic processes

Jessica Leoni [*], Simone Gelmini, Giulio Panzani, Mara Tanelli

Politecnico di Milano, Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Milan, Italy

ARTICLE INFO

Keywords:
Explainable machine-learning
Anomaly detection
Fault detection
Dynamical systems
Features extraction

ABSTRACT

Detecting faults in dynamic systems is challenging due to temporal dependencies and signal correlations. Feature extraction from time-series data is a common step in fault detection, which is usually performed according two main approaches: knowledge-based and statistical. Knowledge-based methods provide interpretable features but require significant design time; also, they usually lack generality. Statistical methods are faster but lead to dimensionality issues and lack of interpretability. To address these challenges, we propose an interpretable and automated feature extraction method. It combines the benefits of knowledge-based and statistical methods, offering a fast solution for extracting effective and interpretable indicators without requiring prior domain knowledge. Also, this method consistently computes a fixed set of interpretable features describing the process's dynamic behavior.

We extensively compare our approach with state-of-the-art methods considering nine open-source datasets spanning various domains. Our results show that classifiers trained on features extracted with the proposed method achieve performance comparable to those provided by state-of-the-art automatic features extractors, according to F1-score, true positive rate, and false positive rate. In addition, our approach proves to be more robust to dimensionality issues and enhances interpretability, extracting a reduced set of features effective at providing insight into the detected anomalies' characteristics. Additionally, we demonstrate that the selected features maintain consistent performance across different classifiers, showcasing their versatility.

1. Problem description and related work

Active process monitoring aims at detecting anomalous behaviors in dynamical systems, which could be indicative of a fault. With advancements in sensing technologies, sophisticated algorithms are now available that rely on a set of informative process variables to monitor and predict its status (Park et al., 2020). These algorithms promote economic savings and safety, help improve system's management and optimization, and prevent damages (Huang et al., 2007). It follows that dynamic process monitoring has become an indispensable part of a wide range of application domains, from industrial plants (Venkatasubramanian et al., 2003) to biological signals (Ukil et al., 2016). However, the unpredictability of faults presents a significant challenge in this endeavor. While it is relatively easy to define the expected behavior of a system, predicting the set of possible faults and their respective trends is much more difficult, especially considering real-case scenarios, where fault-related data is limited. Furthermore, faults can come in different forms, including abrupt, incipient, and intermittent events, each requiring specific approaches for detection (Isermann, 2005). This adds to the complexity of designing effective anomaly detection algorithms. Lastly, the presence of superimposed noise in the measured system variables makes it essential to manage it carefully to avoid misleading results.

In the literature, several anomaly detection methods have been proposed and can be divided into three main groups: knowledge-based, model-based, and data-based, as defined in Miljkovic's taxonomy (Miljković, 2011). Table 1 provides a summary of the strengths and weaknesses of each category, which are further described in the following. Knowledge-based methods rely on predefined decision rules crafted by domain experts, applied either directly to system variables or to synthetic indicators derived from them. These methods are highly interpretable, but lack generality (Angeli, 2010).

In contrast, model-based techniques utilize the principle of analytical redundancy (Isermann, 1984) to detect faults by comparing simulated and measured process variables or synthetic indicators (Youssef et al., 2013). While these methods ensure interpretability, they are

[*] Corresponding author.
E-mail addresses: jessica.leoni@polimi.it (J. Leoni), simone.gelmini@polimi.it (S. Gelmini), giulio.panzani@polimi.it (G. Panzani), mara.tanelli@polimi.it (M. Tanelli).

https://doi.org/10.1016/j.engappai.2023.107510
Received 19 July 2023; Received in revised form 2 November 2023; Accepted 12 November 2023
Available online 15 November 2023

Table 1
Strengths and weaknesses of anomaly detection methods.

Approach	Knowledge Based	Model Based	Data Based
Human Interpretable	✓	✓	✗
Design Time	High	High	Low
General	✗	✗	✓
Domain Expertise	Required	Required	Not Required
Sample Size	Low	Low	High
Complex Systems Compliant	✗	✗	✓

not general as they require the design of a specific model for each considered process.

An example is provided by the work proposed in Djordjević et al. (2022), which presents a model-based approach for estimating faults in steer-by-wire vehicle systems. The proposed approach involves a complex transformation of the nonlinear system model to jointly estimate sensor and actuator faults. While effective and accurate, this method requires significant domain knowledge and design effort. Moreover, in scenarios characterized by highly nonlinear dynamics, model-based techniques are often limited by a trade-off between interpretability and model reliability, potentially leading to increased computational costs and time.

On the other hand, data-based methods analyze time-series describing system variables to learn the statistical distribution of nominal process behavior and detect deviations from it. Since they do not rely on prior system knowledge, data-based approaches are the most general. They require minimal setup effort when applied to new processes, as compared to the other two categories. Occasionally, they can be combined with model-based approaches to enhance interpretability, which is often lacking in pure data-driven methods. This integration strikes a balance, offering solutions that require less design time than fully model-based approaches while being more interpretable than purely data-driven ones, albeit at the expense of generality. For example, in Wang et al. (2023) is proposed an advanced iterative control learning scheme for actuator fault detection, integrating Q-learning into the iterative control learning process. Despite achieving good detection performance, this approach still assumes knowledge of the process model, limiting its generality.

Among data-based methods, control charts have been widely used in univariate and multivariate applications (Shewhart and Deming, 1986) to monitor the temporal behavior of the system variables and detect faults. Within this context, multivariate charts have shown promising results, *e.g.,* MUltivariate CUMulative SUM charts, developed to monitor the residual between the mean trend of the observed process realization and the nominal condition (Crosier, 1988). In addition to control charts, methods are available to identify faults in the spectral domain, which have been proven critical in fault detection problems (Yang et al., 2003). For instance, the one-class cepstrum method in Gelmini et al. (2019) relies on the Martin distance (Martin, 2000) between the cepstral coefficients computed from the nominal system variables and those calculated from the measured ones to inspect for anomalous patterns in the signals' spectrum. While this approach has shown promising results in different application domains, it is not always robust when applied to non-stationary signals.

In the last decades, machine-learning (ML) and deep-learning (DL) have made impressive progress in the field of dynamic process monitoring (Choi et al., 2021). Indeed, those data-based methods can consider the system variables both in time and frequency domain, effectively processing multiple input variables simultaneously. By applying complex, non-linear transformations, they extract meaningful features from input data, representing instances in a dimensional space that facilitates anomaly detection. In terms of learning approaches, ML and DL algorithms can be categorized as either supervised or unsupervised. Supervised methods rely on labeled data to differentiate between nominal and anomalous patterns, while unsupervised methods categorize instances based on their internal structures. Unsupervised methods are commonly preferred for anomaly detection, especially one-class methods, as the datasets often suffer from class imbalance and lack of labels (Choi et al., 2021). However, when a well-defined set of instances representing both nominal and anomalous behaviors is available, supervised methods tend to achieve better results (Zong et al., 2018). Recent supervised approaches, such as the one presented in Roy and Bhaduri (2023), have demonstrated exceptional effectiveness in recognizing anomalies in the context of multi-class classification problems. This paves the way for algorithms capable of distinguishing various fault categories, offering valuable insights. Nonetheless, a significant limitation of this approach is its requirement for adequate data volumes for each anomaly class. Consequently, to date most of the fault detection approaches focus on binary problems, where the classes represent nominal and anomalous process behavior. Within this context, research has shown that convolutional (Choi et al., 2022) and long-short term memory (LSTM) neural networks are promising for time-series supervised anomaly detection (Hundman et al., 2018), while LSTM autoencoders (AE LSTM) lead the way for unsupervised methods (Park et al., 2018). LSTMs are suitable for capturing long-term dependencies in the time-series data and are particularly effective in detecting anomalies in sequences with complex and non-linear temporal relationships between variables. Despite the impressive accuracy attained by these approaches, it is crucial to note that they can be computationally intensive and may necessitate substantial training data. Additionally, their predictive processes are often less interpretable when compared to knowledge- and model-based techniques (Li et al., 2022).

A specific research line addresses this issue, which defines procedures and methods to promote ML and DL models interpretability (Lei et al., 2020). Indeed, despite their robustness and performance, end-users still prefer knowledge- and model-based methods as they can understand the decision-making process of these models (Molnar, 2020; Brito et al., 2022). Interpretable models allow for investigating detected damages' causes, contributing to identifying false alarms, and promoting a better understanding of the monitored process (Du et al., 2019). Furthermore, the implementation of a system on an actual vehicle necessitates certification, which presents an additional challenge when employing ML and DL-based methodologies, as they must be interpretable and explainable to meet the stringent requirements of the certification procedure. To this extent, methods such as Shapley Additive Explanations (Lundberg and Lee, 2017) and Gini Importance (Ceriani and Verme, 2012) metric have been developed to reconstruct the prediction process and make it more transparent and interpretable.

These approaches provide insights into the classification decision logic by ranking features based on their importance in the predictive process. Consequently, the most important features are those that exhibit the most significant differences between nominal and anomalous conditions. However, for these methods to be effective, the extracted features must be limited and related to the process behavior. It turns out that a crucial aspect to consider in the design and development of interpretable data-based models is the features extraction phase (Cantú-Paz, 2004).

Features extraction is the process of transforming the measured time-series into concise indicators. These indicators are designed to capture the temporal dependencies and patterns inherent in the system, potentially resulting in more effective fault detection compared to using the original time-series data. Within the literature, feature extraction methods typically fall into two primary categories: knowledge-based and statistical approaches. Considering the first category, an outstanding example is provided in Tao et al. (2023), where a specialized approach is designed for detecting faults in rolling bearings. This method relies on a deep understanding of the system's dynamics and involves wavelet packet decomposition, reconstruction, and the extraction of energy eigenvectors from sub-bands. These steps result in a 2-D time-frequency map of fault-related features, showcasing the power of knowledge-based design in enhancing feature interpretability, a crucial aspect for building user trust in detection system predictions. However, it is essential to recognize that knowledge-based feature engineering also has its limitations. Indeed, it tends to be highly domain-specific and relies on expert knowledge to formulate the synthetic indicators used for feature extraction. Therefore, while effective within their specific domains, these features often lack generality and can be time-consuming to design.

To address this need, statistical feature extraction algorithms have been developed to automatically derive a concise yet informative set of features (Cantú-Paz, 2004). These algorithms typically rely on capturing variations in system behavior over time and correlations between variables, thereby enhancing the accuracy of anomaly detection models, while minimizing the demand for domain-specific knowledge (Barandas et al., 2020). One prominent example in this domain is TSFEL, which extracts over 60 features across temporal, statistical, and spectral domains for each input variable. However, despite its generality, TSFEL generates numerous features, which can impact interpretability and potentially lead to overfitting. This issue arises because it produces a distinct set of features for each monitored time-series within the original process, which may lead to the curse of dimensionality problem (Köppen, 2000).

Therefore, the aforementioned literature analysis reveals that accurate fault detection models have been developed for dynamic process monitoring, which can be knowledge-, model-, or data-based. While ML and DL approaches demonstrate high accuracy and generality, they are often affected by poor interpretability. To enhance interpretability, it is possible to extract process-related features. Currently, these features are extracted using two primary methods: knowledge-based, which requires more design time, or statistical-based, which may compromise interpretability. Typically, the trade-off between interpretability and design time leans toward the former option. These considerations underscore the need for a rapid, interpretable, domain-independent, and robust feature extraction method tailored specifically for dynamic process monitoring. Indeed, such an algorithm, by harnessing the strengths of both knowledge-based and statistical feature design, would deliver interpretable features with minimal design overhead.

1.1. Novel contributions

In light of the previous considerations, this work presents DIP: a Derivative, Integral and Proportional features extraction approach that provides a general and interpretable set of indicators to enable effective dynamic process monitoring. The key innovation of DIP lies in combining the interpretability which characterizes knowledge-based feature design, with the generality inherent in statistical feature extraction methods. Moreover, DIP is engineered to maintain a fixed feature size, regardless of the number of time-series involved in the monitored process, thereby mitigating the curse of dimensionality. Last, DIP is also extremely versatile. Indeed, the extracted features are specifically tailored to maximize the difference between nominal and anomalous process distributions. This allow DIP to be compliant with a wide range of classification techniques.

Specifically, DIP is designed as a two-stage feature extractor. Stage I involves sensor fusion, merging all provided the time-series into a single signal; Stage II is dedicated to feature computation, and extracts DIP features from this combined signal by means of two filters. This architectural design empowers DIP to consistently extract only three features, regardless of the number of input time-series, ensuring robustness against the curse of dimensionality. The combination of these features effectively captures information across various frequency ranges, presenting a comprehensive overview of the process dynamics. Consequently, the extracted features provide valuable insights into which frequency bands predominantly characterize the dynamics of anomalies within the considered process. This enhances the interpretability providing additional knowledge about the process behavior, and enriching its understanding. More specifically:

- The derivative feature is produced by high-pass filtering the signal. This procedure emphasizes the fast-varying components of the time-series, as obtained by calculating the first-order time derivative of the signal. For this reason, this feature is defined as *derivative* and has the purpose of providing the classifier, instant by instant, insights on the most likely future variation of the process;
- The proportional feature is a scaled version of the signal itself, and allows the classifier to know the process' current state;
- The integral feature is obtained by low-pass filtering the combined signal. Similarly to an integral action, a low-pass filter emphasizes the slow components of the process. In this way, importance is given to the past history.

To maximize the effectiveness of the DIP extracted features when integrating them into any chosen classifier, we have leveraged Kullback–Leibler (KL) divergence (Kullback, 1997) in optimizing filter parameters. Accordingly, they have been meticulously tuned to maximize the dissimilarity between the feature distributions extracted from nominal and anomalous process realizations. This design choice constitutes another DIP strength, as it provides effective features regardless of the specific classification technique employed. Besides, the automatic filters fine-tuning process allow DIP to automatically select the most effective frequency bands of interest for any new process. Consequently, DIP operates without requiring prior knowledge of the process domain. It follows that DIP's design provides flexibility to be adapted to different dynamic processes, allowing it to be used in various domains and for any type of anomaly, *i.e.*, abrupt, incipient, and intermittent.

DIP's effectiveness has been evaluated using 9 open-source datasets associated with dynamic process monitoring across various application domains. To comprehensively assess its generality and versatility, we employed 10 different classifiers, encompassing statistical, dynamic-based and machine-learning approaches. We compared the performance of these classifiers when trained on features extracted by DIP with that of the same classifiers trained using features extracted by a state-of-the-art statistical feature extractor, TSFEL, as well as when trained directly on the original time-series. The results demonstrated that DIP features consistently deliver high detection performance, comparable to TSFEL, while providing more interpretable features. Moreover, DIP prove to be more robust to the curse of dimensionality, maintaining its performance as the number of input time-series increases, unlike TSFEL. Additionally, the performance of various classifiers using DIP features remains consistent, highlighting its compliance with different classification technique. Interpretability was also evaluated, with DIP's features providing valuable insights into the dynamics of anomalies.

The rest of the paper is organized as follows: Section 2 details the stages composing the DIP features extraction process. Then, Section 3 describes the evaluation procedure adopted to evaluate DIP performance and compare it to TSFEL. The achieved results are presented and discussed in Section 4. Last, Section 5 summarizes the main achievements of the proposed work and hints to possible future developments.

2. DIP methodology

As reported in Fig. 1, DIP consists of a two-stage features extraction method. This Section details these two stages: sensors fusion and features computation. The first aims at merging the measured time-series in a single combined signal, and its main purpose is to prevent the curse of dimensionality problem. The latter, instead, identifies the high- and low- frequency ranges that better characterize the fault dynamic and extracts them from the combined signal, producing a reduced set of interpretable features.

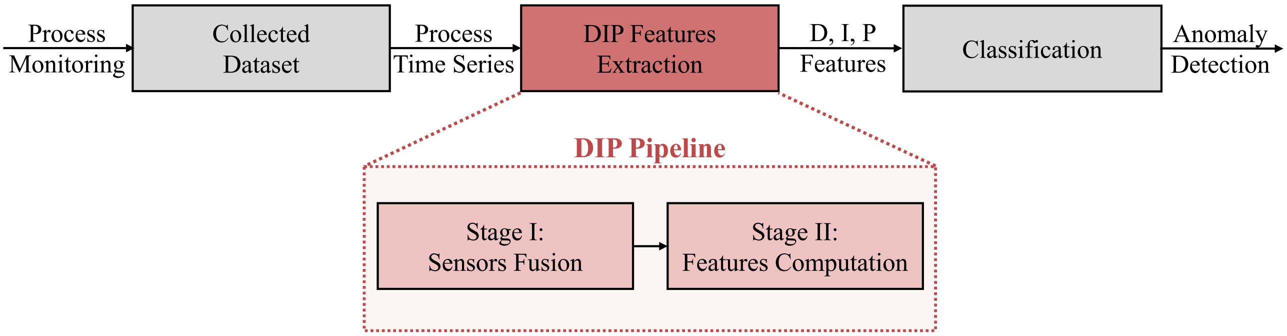


Fig. 1. DIP Pipeline. This Figure shows the pipeline that leads to anomaly detection in a dynamic process. DIP stages are reported in the red squares.

2.1. Stage I: Sensors fusion

Process monitoring involves the acquisition of synchronous time-series data from multiple sensors that measure the relevant variables. Most feature extractors methods, such as TSFEL, calculate a set of features from each time-series, thus increasing the dimensionality of the input space, and potentially incoming to the curse of dimensionality. To address this challenge, the first stage of the DIP method merges the time-series into a single combined signal. Pre-processing is necessary to ensure a consistent merging that represents all the series, including frequency resolution equalization and range normalization.

Frequency resolution equalization is crucial because different sensors may have different sampling frequencies. To address this issue, time-series are resampled so that a fixed time range for each series has the same number of samples. Two commonly used approaches to achieve this goal are downsampling and interpolation. Downsampling reduces the sampling frequency of all the series to the lowest, while interpolation increases the sampling frequency to the highest. In the design of DIP, we adopt time-series interpolation as it maintains all collected samples and adds new samples to the low-sampled series, which is critical in a fault detection scenario where abrupt variations may be present. Furthermore, interpolation guarantees that there are no missing values in the series. At the end of this step, the dataset can be represented as $X = \{\chi_1, \chi_2, \dots, \chi_P\}$, a $\mathbb{R}^{[NxP]}$ matrix composed of set of P time-series, each one composed of N samples. Therefore, each $\chi_j = \{\chi_j(1), \chi_j(2), \dots, \chi_j(N)\}, forj = 1, \dots, P$ is a time-series in $\mathbb{R}^N$.

Before merging the time-series, it is necessary to normalize their range to prevent the series with larger ranges from dominating the others. Several normalization techniques have been proposed in the literature, including min–max normalization, standard normalization, and robust normalization (Ahsan et al., 2021). Min–max normalization maps each sample of a time-series to the interval $[0, 1]$ by subtracting the minimum value of the series and dividing the result by the difference between the minimum and maximum value of the series. The standard normalization, on the other hand, maps each sample of a time-series to a standard Gaussian distribution $\chi_j \sim \mathcal{N}(0, 1).$, by subtracting the mean of the series and dividing the result by its standard deviation. Robust normalization, also known as median scaling, uses the median and the interquartile range to depolarize the distribution of each time-series and adjust its scale. In the design of DIP, robust normalization was chosen as it is less sensitive to outliers and helps ensure that the extracted features from the merged time-series accurately reflect the behavior of the underlying system. Therefore, each χ_j series in the dataset is subjected to:

$$\tilde{\chi}_j(k) = \frac{\chi_j(k) - p_{0.50}(\chi_j)}{p_{0.75}(\chi_j) - p_{0.25}(\chi_j)} \tag{1}$$

where $p_{0.50}$, $p_{0.25}$, and $p_{0.75}$ are the median, first quartile, and third quartile, respectively. To prevent the scaling procedure from being affected by anomalous trends in the data, the quantiles are learned only once from a subset of the data, referred to as the nominal process behavior. In more detail, let α and β be the subsets of the collected data referring to the nominal and fault conditions, respectively. We define $\hat{\alpha}$ as the first $M = 0.75 \cdot \min(N_\alpha, N_\beta)$ samples of each time-series in α. The robust scaling parameters $\Theta = \theta_0, \theta_1, \dots, \theta_P$ are then estimated based on $\hat{\alpha}$ by computing the quantiles $p_{0.25}(\hat{\alpha_j}), p_{0.50}(\hat{\alpha_j}), p_{0.75}(\hat{\alpha_j})$ for each time-series j, and used to normalize the entire dataset.

After normalization, the time-series can be merged. The DIP method involves the summation of synchronous series, taking advantage of the linearity property in the filtering process in the following stage. To maintain the method's generalization capabilities, each time-series is given the same weight during the summation process. However, the approach may also support a non-even weighting of the series based on a ranking that can be either estimated using data-driven techniques or specified a priori.

2.2. Stage II: Features computation

This stage aims to identify and extract from the combined signals the high- and low-frequency bands that are more effective in enhancing the presence of a fault, whether its dynamic is fast or slow. DIP aims to automatically retrieve these bands without resorting to the Fourier Transform, as it is known not to be reliable when applied to non-stationary signals, which is the case of anomalous trends characterized by abrupt transients and outlying behaviors. Therefore, DIP applies a pool of ad-hoc optimized filters to the combined signal to compute its derivative, integral and proportional features.

In this preliminary phase, two first-order filters are considered, a high-pass and a low-pass one. However, the method can be extended to more complex filters. Analytically, the high-pass filter is defined as:

$$H_D(z) = \frac{2\pi f_D(z-1)}{T_s \pi f_D(z+1) + (z-1)},$$

and the low-pass filter is defined as:

$$H_I(z) = \frac{1}{1 + \frac{4\pi}{T_s}\frac{(z-1)}{(z+1)} f_I}$$

where T_s is the time interval between two consecutive samples, and f_I and f_D are the filters' cut-off frequencies.

The high-pass filter aims at extracting the frequency content above f_D, representing the signal's fastest variations. Conversely, the output signal emphasizes the input derivative behavior, providing insights about the most likely future evolution of the process. From the high-pass combined signal the derivative features is computed as:

$$D(k) = (H_D(z) \cdot \psi(k))^2. \tag{2}$$

From a mathematical point of view, it is analogous to computing the combined signal moving variance. The square operator is introduced to prevent the derivative feature from being affected by the combined signal sign. Indeed, only the magnitude is relevant in determining a fault occurrence, as it corresponds to a high process variation from its main trend.

On the other hand, the low-pass filter isolates the harmonics in the combined signals that are slower than f_I, describing the integral process behavior, accounting for its historical evolution. The low-pass combined signal is used to compute the integral feature as:

$$I(k) = H_I(z) \cdot \psi(k). \tag{3}$$

Last, the proportional features is constituted by a dilatation of a factor γ of the proportional signal itself, and describes the process' current behavior, instant by instant.

$$P(k) = \gamma\psi(k). \tag{4}$$

Overall, the Derivative, Integral, and Proportional features provide the classifier with a global vision of the monitored process' temporal trend, knowing its history, present behavior, and the most likely future evolution.

Brent's optimization technique is employed to optimize the filters' frequencies to guarantee that the extracted features are optimal in separating nominal and fault classes. This method, proposed by Brent in 1973, smartly combines the secant method, which guarantees convergence to a solution, with the bisection method, which speeds up the computation. To ensure a feasible frequency to be identified, compliant with the requirements provided by the *Sampling Frequency Theorem* (Farrow et al., 2011), each filters' cut-off frequency was constrained in the range $\mathbb{B} = [0, \frac{f_s}{2}]$. The Kullback-Liebler (KL) divergence was chosen as the cost function (Kullback, 1997). This metric allows for estimating the separation between a reference distribution and another one. It is powerful as it requires a single assumption to be fulfilled to be applicable and robust *i.e.,* that the reference distribution's support is included in that of the other distribution. It should be noted that this is a divergence and not a distance, as this metric is not symmetrical. It follows that the reference distribution choice affects the divergence outcome. In this application, the nominal distribution was considered as the reference.

Therefore, the optimization process is conducted by evaluating the possible cut-off frequencies $(\hat{f}_D, \hat{f}_I)$, and selecting the ones that maximize the divergence between the nominal and fault distribution in the corresponding $\hat{D}$ and $\hat{I}$ features. This design choice guarantees that the optimal parameters set (f_D, f_I) correspond to the filters that extract the D and I features that maximize class-separability, thus easing the prediction task of the classifier, regardless of the specific one employed (Chatterjee and Roychowdhury, 1997). To prevent overfitting and guarantee robustness, the optimization process involves a balanced subset of the data, composed of M nominal and M fault-related samples. From these set, the KL-Divergence is be computed for any derivative and the integral features extracted, ($\hat{f}_D$ and $\hat{f}_I$), as:

$$D_{KL}(D_\alpha \parallel D_\beta) = \sum_{k=1}^{M} D_\alpha(k) \log_2 \left(\frac{D_\alpha(k)}{D_\beta(k)} \right), \tag{5}$$

and

$$D_{KL}(I_\alpha \parallel I_\beta) = \sum_{k=1}^{M} I_\alpha(k) \log_2 \left(\frac{I_\alpha(k)}{I_\beta(k)} \right). \tag{6}$$

where $\hat{D}_\alpha$, $\hat{D}_\beta$, and $\hat{I}_\alpha$, $\hat{I}_\beta$ are the distributions of nominal and fault-related samples in the derivative and integral features, respectively.

To allow the reader for better understanding the whole DIP pipeline, a graphical representation is reported in Fig. 2. Please notice that classification stage is reported in a dashed box, as it is not part of the proposed approach; The DIP approach, in fact, can be employed with any classification method.

3. Evaluation procedure

The DIP features guarantee both interpretability and high detection performance, regardless of the employed classifier or the domain of the process being monitored. To evaluate these strengths, 9 datasets from various application domains were used. The DIP and TSFEL feature sets and the original time-series data were used to train and evaluate the detection performance of 10 different classification techniques. This validation setup was specifically designed to showcase the strengths of the DIP features, including their versatility across different domains, their effectiveness in improving detection compared to the original signals and TSFEL features, and their robustness across various classification techniques.

To this end, this Section first describes how to apply DIP on a given process. Then, it details the datasets used in the evaluation, which are all open source and referenced in the article to enable results reproducibility. Also, it presents the considered classifiers and it explains the performance metrics employed for evaluation purposes.

3.1. DIP applicability

The discussed versatility of DIP enables its application to dynamic processes across various application domains and extract features that provide good detection performance to a wide range of classification techniques commonly employed in process monitoring.

To apply DIP to a given process, an initial fine-tuning phase is necessary, requiring data from both nominal and anomalous process behaviors. Nominal data is used to extract key statistics, namely, the median, the first and the third quantiles, which are required for Stage I in computing the combined signal. Subsequently, both nominal and anomalous process data are essential for fine-tuning the parameters of the low-pass and high-pass filters used in Stage II feature extraction. This fine-tuning process relies on the KL-divergence between feature distributions obtained from nominal and anomalous process samples, ensuring that DIP's features effectively capture the most distinctive patterns between these conditions. Once these parameters are estimated, DIP is ready to be applied to any new acquisition related to the considered process. Naturally, the classifier must also be trained, but the data requirements for training depend on the specific model chosen by the end-user.

Once trained, the DIP stages involve low computational intensity for application, primarily comprising the combination and filtering processes. The computational time and cost of prediction are instead contingent on the chosen classifier.

3.2. Considered dataset

The robustness of DIP is evaluated by considering 9 different datasets from diverse application domains. The datasets have different types of faults affecting different frequency bands and varying numbers of time-series and sampling frequencies. Also, class balance differs among the datasets. A description of each dataset is reported below, while a comparison of their key characteristics is presented in Table 2.

3.2.1. Anxiety

The *anxiety* dataset, discussed in Ihmig Frank et al. (2020), pertains to a biomedical context and includes recordings from 57 spider-fearful volunteers watching a spider video clip. For this analysis, only the recordings of the first volunteer is considered. It consists of 3 time-series, sampled at 100 Hz, measuring the volunteer's electrocardiogram, galvanic skin response, and respiration. Each series is composed of 23 978 samples. Also, a label is reported, indicating the psychological status of the volunteer over time. Accordingly, the first two-thirds of the recording represents its baseline, while the last third, approximately the 33.27% of the whole data, refers to the volunteer's anxiety state induced by the playing of the spider video clip.

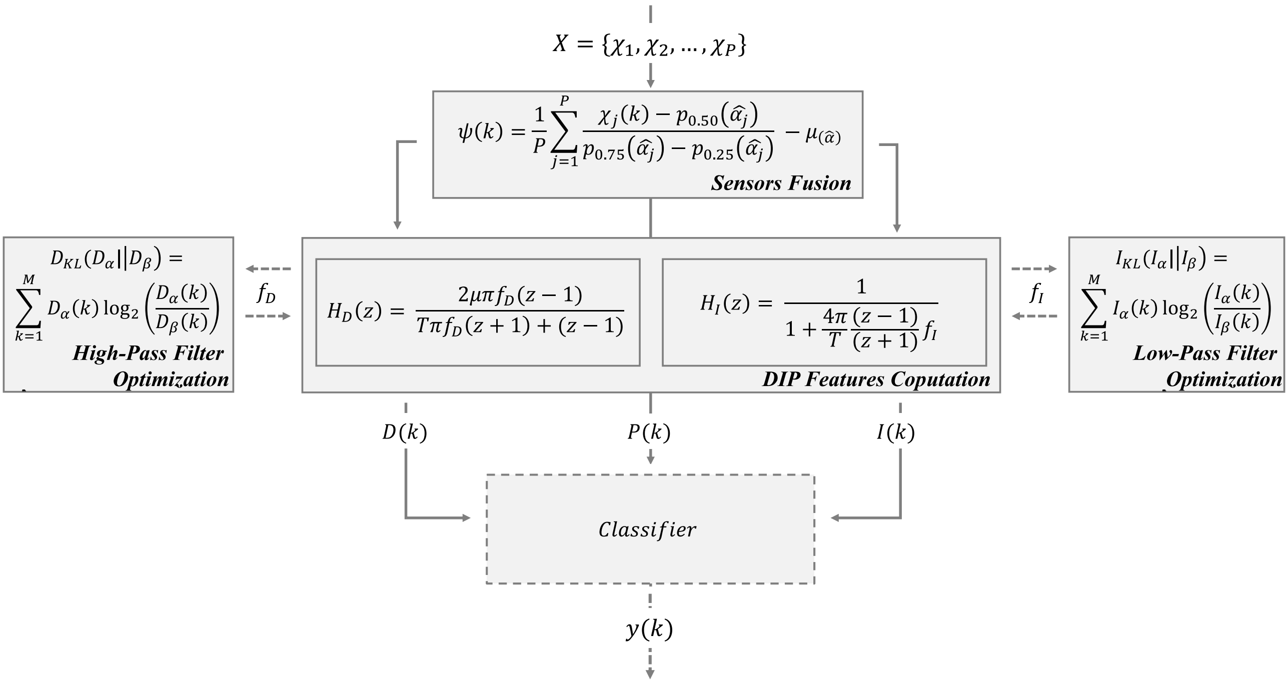

$X = \{\chi_1, \chi_2, \dots, \chi_P\}$
$\psi(k) = \frac{1}{P}\sum_{j=1}^{P} \frac{\chi_j(k) - p_{0.50}(\hat{\alpha}_j)}{p_{0.75}(\hat{\alpha}_j) - p_{0.25}(\hat{\alpha}_j)} - \mu_{(\hat{\alpha})}$
Sensors Fusion
$D_{KL}(D_\alpha \| D_\beta) = \sum_{k=1}^{M} D_\alpha(k) \log_2\left(\frac{D_\alpha(k)}{D_\beta(k)}\right)$
High-Pass Filter Optimization
f_D
$H_D(z) = \frac{2\mu\pi f_D(z-1)}{T\pi f_D(z+1) + (z-1)}$
$H_I(z) = \frac{1}{1 + \frac{4\pi}{T}\frac{(z-1)}{(z+1)} f_I}$
DIP Features Coputation
f_I
$I_{KL}(I_\alpha \| I_\beta) = \sum_{k=1}^{M} I_\alpha(k) \log_2\left(\frac{I_\alpha(k)}{I_\beta(k)}\right)$
Low-Pass Filter Optimization
$D(k)$
$P(k)$
$I(k)$
Classifier
$y(k)$

Fig. 2. DIP pipeline. This figure illustrates the DIP method pipeline, composed of two stages: Stage I, sensors fusion, and Stage II, features computation.

Table 2
Dataset description. This Table describes the datasets used to validate the proposed method.

Dataset	Time series [#]	Duration [s]	fs [Hz]	Anomalous samples [#](%)	Applicative domain
Anxiety	2	250.00	100	7978 (33.27)	Biomedical
e-Call	5	3248.50	100	32 000 (9.85)	Automotive
Letters	3	314.00	200	648 (51.6)	Calligraphy
Lorenz	3	30.00	200	3000 (50.00)	Physical
Occupancy	4	20560.00	0.02	4750 (23.1)	LogistiL
SKAB	1	100.43	100	3524 (35.09)	Mechanical
SWaT	25	946719.00	1	54 621 (5.76)	HydrauliL
Tennessee	50	180.00	10	334 (44.01)	Chemical
WADI	68	192801.00	1	9831 (5.09)	HydrauliL

3.2.2. *e-Call*

The e-Call dataset collects time-series data from 5-axes IMU sensors mounted on motorcycles, acquired during both normal and accident scenarios. The dataset is produced by merging two datasets, described in Gelmini et al. (2019) and Espié et al. (2013). It includes acceleration and pitch and roll rate data, sampled at 100 Hz, and a label indicating the time of the accident. Each series contains 342 850 samples, with 9.85% of them related to accidents.

3.2.3. *Letters*

The *letters* dataset was taken from the character trajectories dataset (Williams et al., 2006). The original dataset consists of 2858 records, each collected by instrumented pen used by a volunteer to write a letter. For each experiment, 3 time series were recorded: the x and y coordinates of the pen tip, and its strength. The sampling frequency of each series was 200 Hz. Each record was labeled with the corresponding letter that was written during the experiment. Since our focus is on anomaly detection and not classification, only the trajectories of two characters, “a” and “e”, were considered. The first was considered the nominal process dynamic, and the latter anomalous. The resulting dataset comprises 62 850 samples for each series, equally distributed between “a” and “e” letters.

3.2.4. *Lorenz attractor*

The *Lorenz attractor* is a complex and non-linear system governed by low-dimensional differential equations that generate chaotic behavior. The considered dataset has been collected from a simulation of this process, and is presented in Kienzler (2018). The dataset is composed of 3 time-series, describing the motion of the system concerning the spatial coordinates. Also, it includes a label to indicate the system’s status over time, *i.e.,* nominal or faulty. All the series were sampled at 200 Hz. Each series in the dataset has 6000 data points, equally balanced between nominal and fault-related.

3.2.5. Occupancy

The *Occupancy Dataset* contains temperature, humidity, light, and CO2 measurements taken in a room to predict the occupancy status of a room that should be occupied most of time (Candanedo and Feldheim, 2016). The variables are sampled at 0.02 Hz and are coupled with a binary label indicating whether the room was occupied. The considered dataset is obtained by concatenating all the recordings available in the data repository and consists of 20 560 data points for each of the 4 measured time-series. This dataset is affected by class unbalance, as only 23.10% of the data points belong to the fault condition.

3.2.6. SKAB

The Skoltech Anomaly Benchmark (SKAB) dataset was ad-hoc collected for evaluating anomaly detection algorithms, and it is referred to a water circulation system testbed. In each experiment, a fault was simulated. To evaluate our approach, we consider the nominal records and those related to valve 1 fault (Katser and Kozitsin, 2020). The dataset is therefore composed of 10 043 samples acquired by a sensor measuring the valve flow rate. Also, a label is provided indicating whether the valve is broken or not. Accordingly, the percentage of data referring to the broken valve is 35.09%.

3.2.7. SWAT

The Secure Water Treatment (SWaT) dataset was created to simulate a water treatment, power generation and distribution, and oil and natural gas refinement plant (Goh et al., 2017). The data was collected over 7 days of normal operation and 4 days of attack scenarios, resulting in 25 time-series each with 946 719 samples. A label indicating the system status is also provided. 5.76% of the samples are reported as related to faults.

3.2.8. Tennessee eastman process

The Tennessee Eastman Process (TEP) is a commonly used benchmark for fault detection in chemical processes (Rieth et al., 2017). The dataset consists of records of 51 variables monitored during a chemical reaction sampled at 10 Hz and labeled as either normal or anomalous. The dataset used in this research was created by combining the first run of the faulty and fault-free training datasets, and includes 51 time series, each 759 samples long. The percentage of anomalous instances is 44.01%.

3.2.9. WADI

The Water Distribution testbed is an extension of the SWaT system and it is designed for water treatment and distribution (Ahmed et al., 2017). The dataset collected from this system consists of 68 time-series, each with 192 801 samples, which describe the behavior of different subsystems. A label indicating the presence of faults is also provided. Only 5.09% of the samples in the dataset are fault-related.

3.3. Selected classifiers

To prove DIP's generality versatility, a classification stage in included in the evaluation pipeline. Specifically, to assess that DIP features effectively provide high detection performance regardless of the specific classification technique, a set of 10 classifiers was chosen, based on different learning paradigms. Details concerning the selected classifiers are summarized in Table 3. The taxonomy is provided according to Brownlee (2013), and allows to group the selected classifiers into three categories:

- ML and DL: This category includes algorithms capable of automatically learning an optimal separation hyperplane to distinguish instances belonging to different distributions. This category can be further divided into:
  - Supervised learning, which learns to separate instances by relying on a set of labeled data. In this work, we considered Random Forest (RF), Logistic Regression (LR), Quadratic Discriminant Analysis (QDA), and LSTM Neural Networks (LSTM). The LSTM comprises an input layer, followed by two LSTM layers of 32 and 16 nodes, respectively. Then a dropout layer is included, with a drop percentage of 30%, to prevent overfitting, and last a dense layer produces a single time-series consisting on the real-time prediction of the process' status.
  - Unsupervised learning, which produces a model to classify instances based on their inner structure. In this work, we considered Isolation Forest (iForest), Local Outlier Factor (LOF), One-Class Support Vector Machines (One-Class SVM), and LSTM Autoencoder (AE-LSTM). The AE-LSTM encoder is composed of an input layer, followed by two LSTM layers of 32 and 16 nodes. Then, a dropout layer follows, with a drop percentage of 30%, which prevents overfitting. The decoder structure is the same as the encoder but symmetrical and ends with a dense layer that reconstructs the output instance.
- Dynamics-based. This approach considers the process dynamics in detecting anomalies, for example, by considering differences in the frequency content behavior between the nominal and fault conditions. In this work, One-Class Cepstrum was considered (Gelmini et al., 2019). It relies on the Martin distance between the cepstral behavior of the nominal system and the measured system. Anomalies are identified when the computed distance exceeds a threshold. This method relies on a single parameter, *i.e.,* the size of the window used to compute the signal spectrum. It was fine-tuned ad-hoc to meet the time-series range in each dataset.
- Control charts. These are statistical tools used to identify anomalies in a process based on a set of rules. In this approach, we considered Multivariate Cumulative Sum Charts (MCUSUM), which monitor the residual between the mean behavior of the observed time-series and that characterizing the nominal condition. An integral of the residual is computed using a fine-tuned sliding window and compared to a threshold. MCUSUM relies on a parameter, k ad-hoc fine-tuned for each dataset to meet the range of the included time-series.

The purpose of leveraging this wide pool of classification techniques to evaluate the effectiveness of our extracted features is twofold. First, it aims to demonstrate that DIP is general across application domains and versatile for most of the classifiers employed in process monitoring. To achieve this, we aim to show that, for each dataset, all classifiers achieve similar performance regardless of their learning paradigm, indicating the intrinsic effectiveness of DIP features. Secondly, this evaluation aims to compare the detection capabilities of DIP features with those provided by the state-of-the-art feature extractor TSFEL and by the original time-series.

To conduct the evaluation, for each dataset, we first fine-tune the classifiers hyperparameters to optimize their performance with the original time-series. Then, the classifiers were trained and evaluated using the original time-series data, TSFEL extracted features, and DIP features as input. To ensure robustness, we compute the performance metric according to a 10-fold cross-validation procedure. This involved training the classifiers on 75% of the data and evaluating their performance on the remaining 25%. This process was repeated $k = 10$ times, with random variations in the composition of the training and test sets, and the average metrics were considered for analysis.

Table 3
Classifiers description. This Table describes the selected classifiers, reporting their hyperparameters, fine-tuned based on the performance assessed by considering the original time-series as input. Parameters that are not specified are set by default.

Classifier	Learning paradigm	Taxonomy	Hyperparameters
Random Forest	Supervised	Bagged ensamble	Estimators = 2 Maximum depth = 3
LogistiL Regression		Regression based	Penalty = elsticnet L1 norm ratio = 0.3; Solver = saga
Quadratic Discriminant Analysis		Dimensionality reduction-based	
LSTM		Multi-layer perceptron	Dropout percentage = 0.3 Loss = binary crossentropy; Optimizer = adam
iForest	Unsupervised	Density based	Estimators = 50 Contamination rate = 0.1
Local Outliers Factor		Distance based	Neighbors = 20 Contamination rate = 0.1, Novelty = True
One-class SVMs		Novelty detection-based	
AE-LSTM		Multi-layer perceptron	Dropout percentage = 0.3 Loss = Root mean squared error; Optimizer = adam
One-class Cepstrum	Frequency Analysis	DynamiL	Window Length = 10s (Anxiety, Lorenz, SkaB, and Tennessee) Window Length = 60s (SWAT, WADI, Letters), 260s (Occupancy), and 100s (e-Call)

MCUSUM	Statistical	Control chart	k = 0.1 (SWAT, WADI, e-Call)
			k = 0.05 (others)

3.4. Evaluation metrics

In evaluating the DIP performance two key indicators have been targeted: the correctness of the predictions and the interpretability of the results. While the literature proposes a lot of metrics to evaluate a classifier's accuracy, assessing the interpretability is challenging. In this work, we evaluate DIP features' interpretability by investigating the frequency ranges identified by the filters optimization stage as optimal to enhance anomalous process behavior. Also, this information is combined with the features' importance ranking produced according to Gini Importance metric.

3.4.1. Predictive capabilities assessment

To evaluate the performances of the selected classifiers in recognizing anomalies 3 metrics have been considered: true positive rate (TPR), false positive rate (FPR), and F1-score. These metrics have been selected due to their robustness with respect to class imbalance, which typically affects anomaly detection datasets. Furthermore, they directly reflect the detection of anomalies and the generation of false alarms, being crucial indicators in dynamic process monitoring.

Specifically, the TPR and FPR are derived from the information contained in the confusion matrix, which is a tool that compares the actual labels with the classifier predictions. The confusion matrix reports four values: true positives (TP), false positives (FP), false Negatives (FN), and true Negatives (TN). True positives represent the number of anomalous instances that were detected the classifier. False positives represent the number of nominal instances predicted as anomalous. False negatives represent the number of anomalous instances that were incorrectly predicted as nominal, while true negatives represent the number of nominal instances that were correctly recognized.

From the confusion matrix TPR, also known as recall (ρ), is calculated as the ratio of the number of true positives to the total number of positive instances.

$$TPR = \frac{TP}{TP + FN} \cdot 100 \tag{7}$$

On the other hand, FPR is calculated as the ratio of the number of false positives to the total number of negative instances.

$$FPR = \frac{FP}{TN + FP} \cdot 100 \tag{8}$$

An high performing classifier should maximize the TRP and minimize the FPR. Last, F1-Score is a more comprehensive measure of performance that takes into account both TPR and FPR, which is defined as the harmonic mean of precision (τ) and recall (ρ). The precision is defined as the ratio between the true positives and the total number of positive predictions reported by the classifier:

$$\tau = \frac{TP}{TP + FP} \cdot 100 \tag{9}$$

Accordingly, the F1-score is computed as:

$$F1 = 2 \cdot \frac{\tau \cdot \rho}{\tau + \rho} \cdot 100. \tag{10}$$

Unlike accuracy, which can be misleading in datasets with unbalanced class distribution, F1-Score provides a balanced view of the performance by considering both precision and recall.

3.4.2. Interpretability assessment

Despite proving that DIP provides predictive capabilities which are comparable to state-of-the-art features extractors, our focus also extends to evaluating the interpretability of the extracted features themselves. However, assessing interpretability can be challenging as objective metrics in this domain are less standardized and often application-specific. Therefore, our evaluation approach involves examining the relationship between the actual dynamics of anomalies and the information provided by DIP-extracted features. To achieve this, we consider two key aspects: first, we analyze the frequency ranges identified as optimal during the cut-off frequency identification process, and second, we assess the importance of each feature in the predictive process. To assess feature importance, we consider the Gini Importance metric (Ceriani and Verme, 2012), which is commonly used in tree-based classifiers as Random Forest to provide an estimate of how much each feature contributes at decreasing impurity in a node after a split, thus

Table 4
F1-score per Learning Paradigm. This Table presents the F1-scores assessed by the selected MLand DLclassifiers when trained and evaluated on DIP, TSFEL, and original time-series, respectively. Scores are aggregated according to the learning paradigm and reported in mean ± standard deviation format.

Features	Classifier	Considered Dataset								
Set	Paradigm	Anxiety	e-Call	Letters	Lorenz	Occupancy	SKAB	SWAT	Tennessee	WADI
DIP	Supervised	99.9 ± 0.1	87.2 ± 4.4	96.3 ± 5.3	99.8 ± 7.2	96.3 ± 1.8	94.9 ± 1.1	95.0 ± 2.4	81.5 ± 5.2	87.6 ± 3.9
	Unsupervised	73.7 ± 13.8	74.8 ± 12.4	64.2 ± 9.8	68.8 ± 13.9	79.0 ± 10.6	69.4 ± 18.3	72.1 ± 12.4	78.3 ± 9.7	76.0 ± 14.8
TSFEL	Supervised	99.4 ± 0.4	74.2 ± 8.4	78.2 ± 7.4	98.8 ± 1.9	87.1 ± 20.1	92.5 ± 0.9	71.3 ± 2.9	91.5 ± 7.5	79.4 ± 13.9
	Unsupervised	65.2 ± 15.5	71.2 ± 15.0	60.8 ± 9.4	67.8 ± 11.0	53.3 ± 46.1	54.7 ± 42.8	71.1 ± 13.7	74.3 ± 7.3	76.6 ± 11.6
Original Time-Series	Supervised	99.9 ± 0.1	85.9 ± 0.3	55.4 ± 8.2	96.8 ± 3.0	98.6 ± 0.3	92.1 ± 0.9	95.1 ± 2.6	78.3 ± 9.2	92.8 ± 0.5
	Unsupervised	72.3 ± 18.6	73.6 ± 14.7	51.5 ± 10.2	72.2 ± 22.9	72.8 ± 17.1	69.5 ± 22.3	71.4 ± 12.6	72.4 ± 9.1	75.9 ± 19.8

increasing class separability. In detail, for the DIP features set extracted from each considered dataset, the Gini Importance metrics is computed during the training process of the Random Forest classifier at each split node as:

$$IG_i = \frac{N_s}{N_t}\left(i - \frac{N_{sr}}{N_s} i_r - \frac{N_{sl}}{N_s} i_l\right), \quad (11)$$

where N_s is the samples number in the considered parent node, N_t is the total number of samples, i is the impurity of the parent node, and the i_r and i_l are those of the right and left child nodes, respectively. Thus, we evaluate DIP features interpretability by combining the information provided by Gini Importance metric and identified frequency ranges. Indeed, the Gini Importance metric reveals which features have played a significant role in the detection process, explaining the classifier decision logic. Meanwhile, the identified frequency ranges provide insights into the specific frequency bands associated with each feature. It follows that, by examining these two sources together, we can understand whether our method effectively guides the classifier to prioritize features that are related to the frequency bands that differ most between nominal and anomalous process behavior.

Additionally, we examine the correlation between the extracted features to ensure that it is not possible to reduce the features further, as each feature provides unique information. To estimate the correlation, Kendall coefficient is considered. Indeed, it is a non-parametric correlation measure, which does not make any assumptions about the statistical distribution of the samples and is robust to outliers. Therefore, for any DIP features pair, the Kendall correlation coefficient is computed as:

$$corr(i,j) = \frac{N_c(i,j) - N_d(i,j)}{\frac{1}{2}N(N-1)}$$

where N_c and N_d is the number of concordant and discordant synchronous samples pair in the i_{th} and j_{th} features, respectively, and N is the time-series length.

4. Results evaluation and discussion

This Section presents and discusses the results of the evaluation of DIP performance in terms of predictive capabilities and interpretability. For each considered process, the comparison of key metrics, such as F1-score, true positive rate, and false positive rate between DIP, TSFEL, and original time-series, was conducted by providing their features to various classifiers. The results show the advantages of using DIP and provide an in-depth analysis of its features' interpretability.

4.1. Predictive capabilities

As reported in Section 3, the evaluation of the predictive capabilities provided by DIP, TSFEL, and original time-series sets was conducted using key metrics, such as the F1-score, true positive rate, and false positive rate. The results were obtained by considering each dataset and providing the feature sets to the selected classifiers, which were trained and tested through a 10-fold cross-validation procedure.

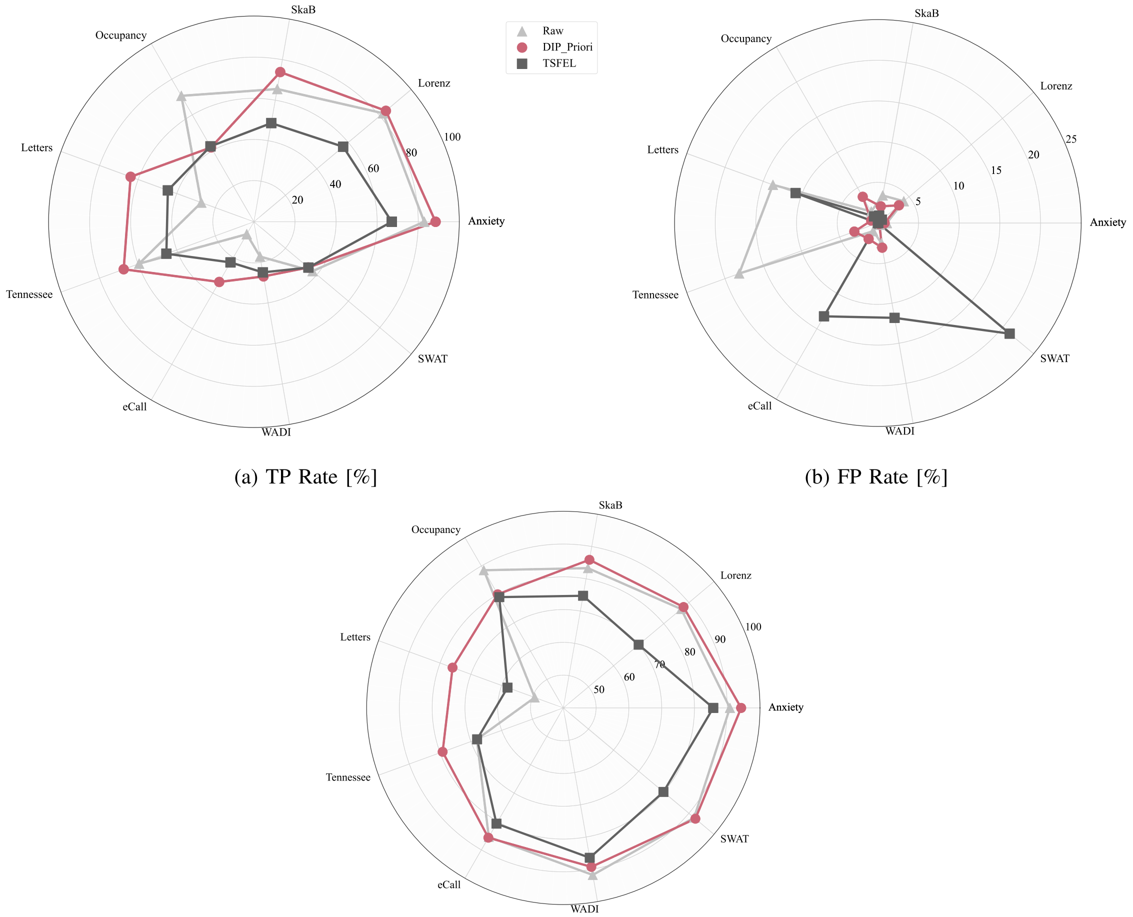
The results, as depicted in Fig. 3, reveal that DIP and TSFEL outperforms the original time-series. Indeed, when compared to original time-series performance, DIP prove to provide higher scores in all the evaluation metrics, with some exceptions where the original time-series lead to slightly higher F1-scores for the *Occupancy* and *WADI* datasets. However, DIP still provides a higher true positive rate in the case of the *WADI* dataset. On the other hand, TSFEL and the original time-series sets experience a drop in performance in certain datasets. TSFEL exhibits a high false positive rate in the case of the *SWAT, WADI*, and *e-Call* datasets. Similarly, the original time-series sets lead to high false positive rates in the case of the *Letters* and *Tennessee* datasets. A high false positive rate corresponds to the number of false alarms triggered, which is a well-known issue in anomaly detection systems, which many researchers aim to overcome. Indeed, false alarms result in time and economic losses due to unnecessary inspections.

The results show that TSFEL's performance is impacted by class imbalance, resulting in poor performance when applied to datasets where the anomaly class is underrepresented. Meanwhile, the original time-series are impacted by class separability, as seen in Fig. 4 which represents the probability distribution of nominal and anomalous classes estimated by the combined signals in *Letters, Tennessee* and *Anxiety* datasets, respectively. It turns out that the classifiers' performance drops in datasets where the distributions overlap, as *Letters* and *Tennessee*. On the other hand, in datasets like *Anxiety*, where the original signals have high class separability, the original time-series allow the classifiers to correctly detect fault-related samples. These results support the design choice of DIP, which is optimized to maximize class separability and simplify the classification task.

The predictive capabilities provided by DIP and TSFEL features, and by the original time-series are assessed in Tables 4 and 5. These tables break down the F1-score results of ML and DL classifiers when trained on each of these features' set. Supervised learning approaches generally outperform other paradigms, while statistical and dynamics-based techniques exhibit similar performance to unsupervised learning methods. Statistical methods excel in handling class imbalances, while dynamics-based methods excel in detecting rapidly changing anomalies. Regardless of the chosen classifier, DIP consistently demonstrates its versatility, being able to provide robust, general, and accurate predictions. Moreover, the results reveal that both DIP and TSFEL features outperform the original time-series data. However, in datasets with a substantial number of time-series, such as *SWAT* and *WADI*, TSFEL faces dimensionality challenges, leading to reduced performance compared to DIP. This highlights that DIP is less affected by the curse of dimensionality, thanks to the signal combination performed in Stage I.

4.2. DIP features interpretability

Despite proving DIP features capability at providing high predictive capabilities, we are interested in evaluating its interpretability. To this extent, we consider the contribution supplied by DIP features to understand the fault behavior in the monitored system and the frequency ranges most relevant to the fault detection task. This can improve trust in the ML and DL models and encourage their wider adoption in industrial settings.

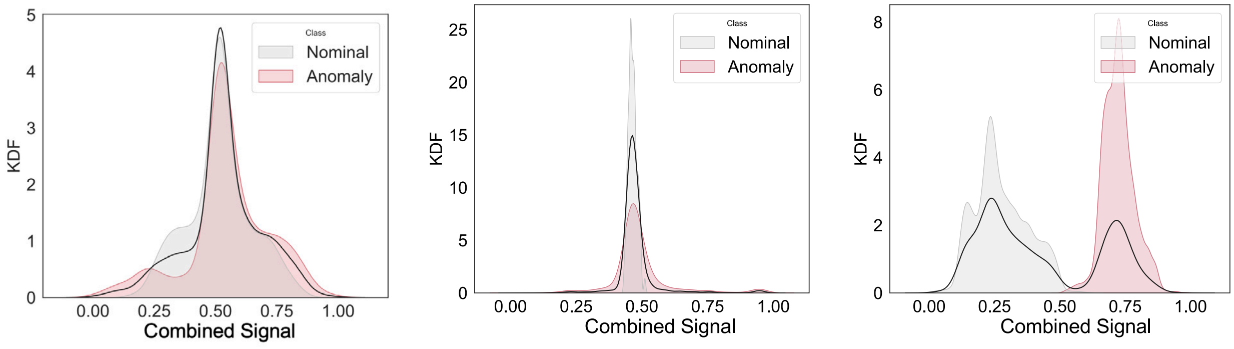


(a) TP Rate [%]

(b) FP Rate [%]

(c) F1-score [%]

Fig. 3. TP rate, FP rate, and delay performances. This Figure reports the average detection performance assessed by the selected classifiers when evaluated on DIP, TSFEL, and original time-series.



(a) Letters

(b) Tennessee

(c) Anxiety

Fig. 4. Class-Separability in *Letters*, *Tennessee*, and *Anxiety* datasets. This Figure shows the probability distribution for the nominal and anomalous classes in the combined signal. It turns out that *Anxiety* provides high separability, while further pre-processing is required in *Letters* and *Tennessee* to differentiate classes' behavior.

As reported in Section 3, interpretability is evaluated by combining the features importance ranking produced considering Gini metric with the information concerning the frequency ranges reported as most important according to the filters optimization process. Table 6 reports the frequency range as optimal during filters' fine-tuning stage. Also Table 7 provides insight concerning the features importance of each extracted feature, estimated according to Gini Importance metric computed during the training process of the Random Forest classifier on DIP features.

The results show that the most important feature for detecting anomalies varies across different datasets. In the *Lorenz*, *Occupancy*, *Tennessee*, and *WADI* datasets, the integral feature is particularly important to provide the classifiers with high detection capabilities. Furthermore, the optimal cut-off frequency for the derivative feature in these datasets is very low, suggesting that it should also capture slow-varying content, thus further emphasizing the importance of the integral feature.

Table 5
F1-score per Classification Category. This Table presents the F1-scores assessed by the selected classifiers' categories when trained and evaluated on DIP, TSFEL, and original time-series, respectively. Please notice that for the MLand DLcategory are reported the outcomes of the best approach, *i.e., Random Forest classifier*. Scores are reported in mean ± standard deviation format.

Features	Selected	Considered Dataset								
Set	Classifier	Anxiety	e-Call	Letters	Lorenz	Occupancy	SKAB	SWAT	Tennessee	WADI
DIP	Random Forest	99.9 ± 0.1	88.5 ± 3.5	98.7 ± 1.1	99.2 ± 0.4	87.1 ± 1.4	94.2 ± 0.3	97.2 ± 0.9	99.8 ± 0.1	91.8 ± 1.9
	MCUSUM	97.2 ± 1.6	85.5 ± 3.1	84.2 ± 4.7	80.1 ± 3.1	66.9 ± 2.8	53.5 ± 7.1	78.9 ± 2.4	91.7 ± 2.8	91.8 ± 1.4
	One-Class Cepstrum	74.6 ± 2.5	83.1 ± 3.7	86.3 ± 3.1	56.38 ± 6.0	65.7 ± 5.7	79.1 ± 3.9	71.2 ± 3.1	51.1 ± 9.2	91.7 ± 3.8
TSFEL	Random Forest	99.9 ± 0.1	90.0 ± 0.8	98.1 ± 0.8	99.9 ± 0.1	98.6 ± 0.6	91.6 ± 0.2	97.2 ± 0.9	85.5 ± 2.1	89.6 ± 1.3
	MCUSUM	81.4 ± 2.6	85.4 ± 2.1	83.3 ± 0.7	75.8 ± 2.9	66.9 ± 6.1	50.9 ± 7.5	79.1 ± 3.7	89.3 ± 2.1	87.4 ± 1.6
	One-Class Cepstrum	81.8 ± 3.8	82.2 ± 3.7	53.3 ± 6.1	68.8 ± 5.6	63.4 ± 6.1	69.4 ± 5.1	72.3 ± 3.1	50.5 ± 9.2	90.5 ± 0.8
Original Time Series	Random Forest	99.9 ± 0.1	86.3 ± 1.2	90.2 ± 0.9	99.9 ± 0.1	88.6 ± 2.7	91.6 ± 2.4	93.0 ± 3.1	83.7 ± 3.8	90.2 ± 2.8
	MCUSUM	93.3 ± 0.7	87.1 ± 4.1	94.2 ± 1.7	83.3 ± 3.1	65.7 ± 8.1	54.3 ± 9.3	72.1 ± 5.3	97.8 ± 0.7	92.4 ± 2.4
	One-Class Cepstrum	95.6 ± 1.2	85.5 ± 4.1	58.9 ± 8.1	50.5 ± 7.1	67.1 ± 7.4	68.1 ± 4.2	68.9 ± 5.7	51.1 ± 11.3	91.3 ± 0.8

Table 6
DIP Derivative and Integral Frequency Ranges. This Table reports frequency range identified as optimal in the filters fine-tuning process to enhance the presence of fault behavior in the analyzed datasets.

Dataset	Frequency range	
	I	D
Anxiety	[0.00–6.09]	[23.03–50.00]
e-Call	[0.00–1.00]	[30.23–50.00]
Letters	[0.00–5.50]	[52.5–100.00]
Lorenz	[0.00–9.10]	[50–100.00]
Occupancy	[0.00–0.01]	[0.01–0.02]
SKAB	[0.00–0.69]	[33.20–50.00]
SWAT	[0.00–0.50]	[0.49–0.50]
Tennessee	[0.00–3.00]	[0.48–5.00]
WADI	[0.00–0.10]	[0.09–0.50]

Table 7
DIP Features Importance. This Table reports the features importance estimated for the DIP features according to Gini Importance metric applied to the Random Forest training process.

Dataset	Features importance		
	D	I	P
Anxiety	0.04	0.59	99.37
e-Call	11.23	39.66	49.11
Letters	56.59	18.85	15.56
Lorenz	8.68	63.51	27.81
Occupancy	1.37	98.22	0.41
SKAB	0.06	53.52	46.42
SWAT	44.75	5.18	50.07
Tennessee	20.80	67.70	11.50
WADI	8.05	63.04	28.91

Table 8
DIP Features Correlation. This Table reports Kendall correlation coefficients for each pair of extracted DIP features.

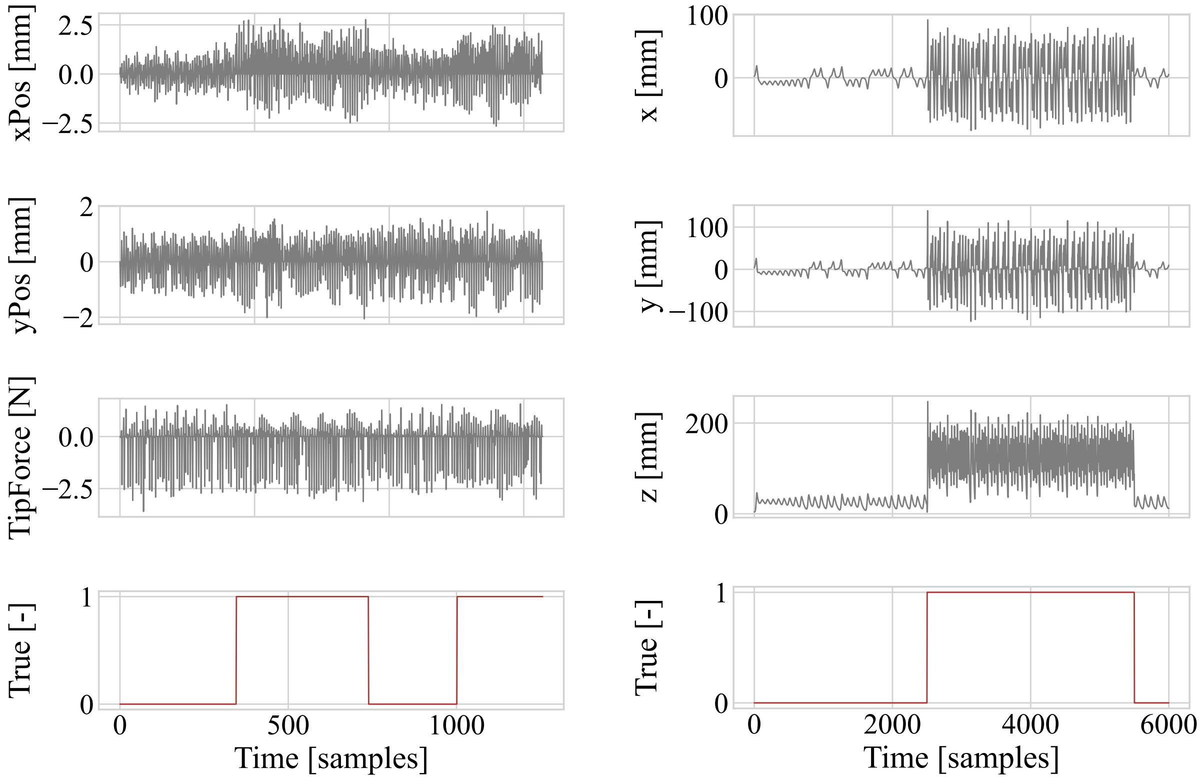
Dataset		Anxiety	e-Call	Letters	Lorenz	Occupancy	SKAB	SWAT	Tennessee	WADI
Kendall Correlation	D-I	0.07	0.04	0.10	0.37	0.07	0.00	0.08	0.11	0.07
	I-P	0.38	0.10	0.11	0.64	0.99	0.11	0.93	0.11	0.61
	D-P	0.05	0.00	0.05	0.31	0.06	0.07	0.08	0.02	0.07

On the other hand, faults in *Letters* process are most characterized by the derivative feature. In this dataset, the optimal cut-off frequency identified for the integral feature is considerably high, further proving the importance of fast-varying trends in enhancing the presence of a fault. Considering *Anxiety* and *e-Call*, the information provided by the combined signal, *i.e.* the proportional feature, is per se effective in detecting faults. In *SKAB* and *SWAT* the information provided by the combined signal is informative but should be considered along with the low- and high-frequency content, respectively, to guarantee high detection capabilities. To provide the reader with a comparison of a process characterized by fast- and slow-varying fault behaviors, *Letters* and *Lorenz* original time-series and the respective label are reported in Fig. 5.

Last, correlation analysis of the DIP features is performed to assess that they provide unique information and are not redundant. The achieved results, reported in Table 8, reveal that in most of the considered datasets, the DIP features are minimally correlated, indicating that they provide distinct information. In the *Occupancy* and *SWAT* datasets only, there is a strong correlation between the integral and proportional features, which is expected due to the high cut-off frequency identified as optimal for the low-pass filter.

5. Final considerations and future work

In this study, we introduce DIP, a dynamic-aware feature extraction method designed to robustly capture a consistent set of interpretable features across diverse processes. DIP's architecture is two-stage: Stage I, sensor fusion, combines multiple time-series into a single signal; Stage II, features computation, extracts low-frequency and high-frequency features using two optimized filters. This configuration allows DIP to be resistant to the curse of dimensionality, extracting a fixed set of features



(a) Letters

(b) Lorenz

Fig. 5. Time-Series Trends in *Letters* and *Lorenz* datasets. This Figure shows the temporal trends of the signals acquired for the *Letters* and *Lorenz* datasets, along with the actual system status label. According to DIP optimization process, in *Lorenz* anomalies most characterize by low-frequency signals behavior, while in *Letters* high-frequencies are most determining.

regardless of the input time series. Furthermore, the filter optimization technique, which aims to maximize the distinction between nominal and anomalous feature distributions, ensures the method's generality and versatility. Indeed, DIP can be applied to new dynamic processes, and the features it generates prove effective when used with a wide range of classification techniques commonly employed in process monitoring. Furthermore, DIP provides features that are more interpretable than those extracted from state-of-the-art approaches. Indeed, Derivative, Integral, and Proportional features provide valuable insights into the dynamics of detected anomalies.

To assess DIP's generality and versatility, we conducted experiments across nine datasets related to different application domains. For each one, DIP features have been extracted and proposed to ten classifiers based on various learning paradigms. We compared the evaluation results with those provided by features extracted by training the classifiers on state-of-the-art method, TSFEL, and directly on the original time-series data. Our findings demonstrated that, while achieving comparable detection performance, DIP outperformed both in terms of robustness and interpretability.

5.1. Limitations and future work

While our feature extraction approach offers generality and interpretability, it comes with inherent limitations. Firstly, in scenarios involving complex processes with numerous time-series data, where the frequency behavior between nominal and anomalous behavior is not so different, knowledge-based features tailored to specific behaviors may outperform DIP. Furthermore, the current version of DIP presents challenges when incorporating a priori knowledge about the process. Indeed, customization options are currently limited to weightings applied during feature combination in Stage I, which may not always offer straightforward fine-tuning. Another limitation is that, as many anomaly detection techniques in the literature, DIP needs both nominal and anomalous process behavior data for proper training. The representativeness of nominal and anomalous classes in the training dataset is crucial for effective anomaly detection, but ensuring a balanced dataset can be challenging in real-world scenarios. Lastly, while DIP filters are designed to be robust against noise, extremely corrupted or time-series data can still pose challenges. In such cases, the extracted features may not effectively recognize anomalies or may lead to false positives.

Future work deal with exploring different architectures for the low-pass and high-pass filters to determine the most suitable one. Additionally, research will be focus on enhancing customization, and analyzing DIP's robustness with respect to noise.

CRediT authorship contribution statement

Jessica Leoni: Conceptualization, Methodology, Software, Formal analysis, Writing – original draft, Visualization. **Simone Gelmini:** Conceptualization, Methodology, Software, Supervision, Writing – review & editing. **Giulio Panzani:** Conceptualization, Methodology, Resources, Writing – review & editing, Visualization, Supervision, Project administration. **Mara Tanelli:** Conceptualization, Methodology, Resources, Writing – review & editing, Visualization, Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Ahmed, C.M., Palleti, V.R., Mathur, A.P., 2017. WADI: a water distribution testbed for research in the design of secure cyber physical systems. In: Proceedings of the 3rd International Workshop on Cyber-Physical Systems for Smart Water Networks. pp. 25–28. http://dx.doi.org/10.1145/3055366.3055375.

Ahsan, M.M., Mahmud, M., Saha, P.K., Gupta, K.D., Siddique, Z., 2021. Effect of data scaling methods on machine learning algorithms and model performance. Technologies 9, 1–17. http://dx.doi.org/10.3390/technologies9030052.

Angeli, C., 2010. Diagnostic expert systems: From expert's knowledge to real-time systems. Adv. Knowl. Based Syst. Model, Appl. Res. 1, 50–73.

Barandas, M., Folgado, D., Fernandes, L., Santos, S., Abreu, M., Bota, P., Liu, H., Schultz, T., Gamboa, H., 2020. TSFEL: Time series feature extraction library. SoftwareX 11, 1–7. http://dx.doi.org/10.1016/j.softx.2020.100456.

Brito, L.C., Susto, G.A., Brito, J.N., Duarte, M.A., 2022. An explainable artificial intelligence approach for unsupervised fault detection and diagnosis in rotating machinery. Mech. Syst. Signal Process. 163, 1–21. http://dx.doi.org/10.1016/j.ymssp.2021.108105.

Brownlee, J., 2013. A tour of machine learning algorithms. Mach. Learn. Mastery 25.

Candanedo, L.M., Feldheim, V., 2016. Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models. Energy Build. 112, 28–39. http://dx.doi.org/10.1016/j.enbuild.2015.11.071.

Cantú-Paz, E., 2004. Feature subset selection, class separability, and genetic algorithms. In: Genetic and Evolutionary Computation Conference. Springer, pp. 959–970.

Ceriani, L., Verme, P., 2012. The origins of the Gini index: extracts from Variabilità e Mutabilità (1912) by Corrado Gini. J. Econ. Inequal. 10 (3), 421–443. http://dx.doi.org/10.1007/s10888-011-9188-x.

Chatterjee, C., Roychowdhury, V.P., 1997. On self-organizing algorithms and networks for class-separability features. IEEE Trans. Neural Netw. 8, 663–678. http://dx.doi.org/10.1109/72.572105.

Choi, T., Lee, D., Jung, Y., Choi, H.-J., 2022. Multivariate time-series anomaly detection using seqvae-CNN hybrid model. In: 2022 International Conference on Information Networking. ICOIN, pp. 250–253. http://dx.doi.org/10.1109/ICOIN53446.2022.9687205.

Choi, K., Yi, J., Park, C., Yoon, S., 2021. Deep learning for anomaly detection in time-series data: review, analysis, and guidelines. IEEE Access 9, 120043–120065. http://dx.doi.org/10.1109/ACCESS.2021.3107975.

Crosier, R.B., 1988. Multivariate generalizations of cumulative sum quality-control schemes. Technometrics 30 (3), 291–303.

Djordjević, V., Stojanović, V., Pršić, D., Dubonjić, L., Morato, M.M., 2022. Observer-based fault estimation in steer-by-wire vehicle. Eng. Today 1 (1), 7–17.

Du, M., Liu, N., Hu, X., 2019. Techniques for interpretable machine learning. Commun. ACM 63 (1), 68–77. http://dx.doi.org/10.1145/3359786.

Espié, S., Boubezoul, A., Aupetit, S., Bouaziz, S., 2013. Data collection and processing tools for naturalistic study of powered two-wheelers users' behaviours. Accid. Anal. Prev. 330–339.

Farrow, C.L., Shaw, M., Kim, H., Juhás, P., Billinge, S.J., 2011. Nyquist-Shannon sampling theorem applied to refinements of the atomic pair distribution function. Phys. Rev. B 84, 1–7. http://dx.doi.org/10.1103/PhysRevB.84.134105.

Gelmini, S., Panzani, G., Savaresi, S., 2019. Analysis and development of an automatic ecall for motorcycles: a one-class cepstrum approach. In: 2019 IEEE Intelligent Transportation Systems Conference. ITSC, IEEE, pp. 3025–3030. http://dx.doi.org/10.1109/ITSC.2019.8916907.

Goh, J., Adepu, S., Junejo, K.N., Mathur, A., 2017. A dataset to support research in the design of secure water treatment systems. In: Critical Information Infrastructures Security: 11th International Conference, CRITIS 2016, Paris, France, October 10–12, 2016, Revised Selected Papers 11. Springer, pp. 88–99.

Huang, S., Tan, K.K., Lee, T.H., 2007. Automated fault detection and diagnosis in mechanical systems. IEEE Trans. Syst. Man Cybern. C (Appl. Rev.) 37, 1360–1364. http://dx.doi.org/10.1109/TSMCC.2007.900623.

Hundman, K., Constantinou, V., Laporte, C., Colwell, I., Soderstrom, T., 2018. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 387–395.

Ihmig Frank, R., Gogeascoechea, A., Schäfer, S., Lass-Hennemann, J., M., T., 2020. Electrocardiogram, Skin Conductance and Respiration from Spider-Fearful Individuals Watching Spider Video Clips. Physionet, http://dx.doi.org/10.13026/sq6q-zg04.

Isermann, R., 1984. Process fault detection based on modeling and estimation methods—A survey. Automatica 20, 387–404. http://dx.doi.org/10.1016/0005-1098(84)90098-0.

Isermann, R., 2005. Model-based fault-detection and diagnosis–status and applications. Annu. Rev. control 29, 71–85. http://dx.doi.org/10.1016/j.arcontrol.2004.12.002.

Katser, I.D., Kozitsin, V.O., 2020. Skoltech Anomaly Benchmark (SKAB). Kaggle, http://dx.doi.org/10.34740/KAGGLE/DSV/1693952.

Kienzler, R., 2018. Lorenz attractor dataset. https://github.com/romeokienzler/developerWorks/tree/master/lorenzattractor.

Köppen, M., 2000. The curse of dimensionality. In: 5th Online World Conference on Soft Computing in Industrial Applications, Vol. 1. WSC5, pp. 4–8.

Kullback, S., 1997. Information Theory and Statistics. Courier Corporation.

Lei, Y., Yang, B., Jiang, X., Jia, F., Li, N., Nandi, A.K., 2020. Applications of machine learning to machine fault diagnosis: A review and roadmap. Mech. Syst. Signal Process. 138, 1–39. http://dx.doi.org/10.1016/j.ymssp.2019.106587.

Li, X., Xiong, H., Li, X., Wu, X., Zhang, X., Liu, J., Bian, J., Dou, D., 2022. Interpretable deep learning: Interpretation, interpretability, trustworthiness, and beyond. Knowl. Inf. Syst. 64, 3197–3234. http://dx.doi.org/10.1007/s10115-022-01756-8.

Lundberg, S.M., Lee, S.-I., 2017. A unified approach to interpreting model predictions. Adv. Neural Inf. Process. Syst., Vol. 30 1–10.

Martin, R.J., 2000. A metric for ARMA processes. IEEE Trans. Signal Process. 48, 1164–1170. http://dx.doi.org/10.1109/78.827549.

Miljković, D., 2011. Fault detection methods: A literature survey. In: 2011 Proceedings of the 34th International Convention MIPRO. IEEE, pp. 750–755.

Molnar, C., 2020. Interpretable Machine Learning. Lulu.

Park, Y.-J., Fan, S.-K.S., Hsu, C.-Y., 2020. A review on fault detection and process diagnostics in industrial processes. Processes 8, 1123. http://dx.doi.org/10.3390/pr8091123.

Park, D., Hoshi, Y., Kemp, C.C., 2018. A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder. IEEE Robot. Autom. Lett. 3, 1544–1551. http://dx.doi.org/10.1109/LRA.2018.2801475.

Rieth, C.A., Amsel, B.D., Tran, R., Cook, M.B., 2017. Additional Tennessee Eastman Process Simulation Data for Anomaly Detection Evaluation. Harvard Dataverse, http://dx.doi.org/10.7910/DVN/6C3JR1.

Roy, A.M., Bhaduri, J., 2023. DenseSPH-YOLOv5: An automated damage detection model based on DenseNet and swin-transformer prediction head-enabled YOLOv5 with attention mechanism. Adv. Eng. Inform. 56, 102007.

Shewhart, W.A., Deming, W.E., 1986. Statistical Method from the Viewpoint of Quality Control. Courier Corporation.

Tao, H., Qiu, J., Chen, Y., Stojanovic, V., Cheng, L., 2023. Unsupervised cross-domain rolling bearing fault diagnosis based on time-frequency information fusion. J. Franklin Inst. B 360 (2), 1454–1477.

Ukil, A., Bandyoapdhyay, S., Puri, C., Pal, A., 2016. IoT healthcare analytics: The importance of anomaly detection. In: 2016 IEEE 30th International Conference on Advanced Information Networking and Applications. AINA, IEEE, pp. 994–997. http://dx.doi.org/10.1109/AINA.2016.158.

Venkatasubramanian, V., Rengaswamy, R., Yin, K., Kavuri, S.N., 2003. A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. Comput. Chem. Eng. 27, 293–311. http://dx.doi.org/10.1016/S0098-1354(02)00160-6.

Wang, R., Zhuang, Z., Tao, H., Paszke, W., Stojanovic, V., 2023. Q-learning based fault estimation and fault tolerant iterative learning control for MIMO systems. ISA Trans.

Williams, B., Toussaint, M., Storkey, A., 2006. UCI Character Trajectories Data Set. School of Informatics, University of Edinburgh, URL https://archive.ics.uci.edu/ml/datasets/Character+Trajectories.

Yang, H., Mathew, J., Ma, L., 2003. Vibration feature extraction techniques for fault diagnosis of rotating machinery: a literature survey. In: Asia-Pacific Vibration Conference. pp. 801–807.

Youssef, A.B., El Khil, S.K., Slama-Belkhodja, I., 2013. State observer-based sensor fault detection and isolation, and fault tolerant control of a single-phase PWM rectifier for electric railway traction. IEEE Trans. Power Electron. 28 (12), 5842–5853. http://dx.doi.org/10.1109/TPEL.2013.2257862.

Zong, B., Song, Q., Min, M.R., Cheng, W., Lumezanu, C., Cho, D., Chen, H., 2018. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In: International Conference on Learning Representations. pp. 1–19.